



# DESIGN OF ATTITUDE CONTROL ACTUATORS FOR A SIMULATED SPACECRAFT

THESIS

Christopher G. McChesney, Captain, USAF

AFIT/GA/ENY/11-M12

DEPARTMENT OF THE AIR FORCE  
AIR UNIVERSITY

**AIR FORCE INSTITUTE OF TECHNOLOGY**

Wright-Patterson Air Force Base, Ohio

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, the Department of Defense, or the United States Government. This material is declared the work of the U.S. Government and is not subject to copyright protection in the United States.

AFIT/GA/ENY/11-M12

DESIGN OF ATTITUDE CONTROL ACTUATORS FOR A SIMULATED  
SPACECRAFT

THESIS

Presented to the Faculty  
Department of Aeronautics and Astronautics  
Graduate School of Engineering and Management  
Air Force Institute of Technology  
Air University  
Air Education and Training Command  
In Partial Fulfillment of the Requirements for the  
Degree of Master of Science in Astronautical Engineering

Christopher G. McChesney, BS  
Captain, USAF

March 2011

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AFIT/GA/ENY/11-M12

DESIGN OF ATTITUDE CONTROL ACTUATORS FOR A SIMULATED  
SPACECRAFT

Christopher G. McChesney, BS  
Captain, USAF

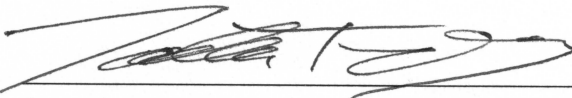
Approved:



Dr. Eric D. Swenson (Chairman)

15 Mar 2011

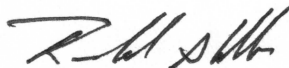
Date



Dr. Jonathan T. Black

15 MAR 2011

Date



Dr. Richard G. Cobb

15 MAR 2011

Date



*Abstract*

The Air Force Institute of Technology's attitude dynamics simulator, SimSat, is used for hardware-in-the-loop validation of new satellite control algorithms. To provide the capability to test algorithms for control moment gyroscopes, SimSat needed a control moment gyroscope array. The goal of this research was to design, construct, test, and validate a control moment gyroscope array for SimSat. A secondary goal of this research was to address known issues with SimSat's existing reaction wheel and thruster control systems. The control moment gyroscope array was required to interface with SimSat's existing structure, power supply, and electronics. The array was also required to meet maneuver specifications and disturbance rejection specifications. First, the array was designed with initial sizing estimates based on requirements and vehicle size. Next, the vehicle and control dynamics were modeled to determine control moment gyroscope requirements and provide a baseline for validation. Control moment gyroscopes were then built, calibrated, and installed on the vehicle. SimSat's existing control issues were addressed during the installation of the control moment gyroscopes. The actuators were then validated against the dynamics model. Testing shows minor deviation from the expected behavior as a result of small misalignments from the theoretical design. Once validation was complete, the control moment gyroscope array was tested against the performance specifications. The performance tests indicated that the control moment gyroscope array is capable of meeting specifications.

## *Acknowledgements*

First and foremost I would like to thank the ENY department and my thesis advisor, Dr. Eric Swenson for allowing me to work on this project. Dr. Swenson was always supportive of my ideas, allowing me enough freedom to make mistakes, then point me in the correct direction. Without his support I'd probably still be stuck at square one. Addition thanks goes to my predecessor Captain Ryan Snider for sharing his lessons about Simsat, practical knowledge, and valuable hands-on time with someone who knew the system.

Next, I'd like to thank the AFIT Model Shop. Brian Crabtree was instrumental in helping me convert my low fidelity design into a complete CAD model ready for production. His inputs minimized the time required by the machine shop to manufacture the components, maximizing my time to address everything else. Without his help my design would have never made it off the paper. I'd also like to thank Dan Ryan and Chris Harkless for their long hours of careful work manufacturing the parts that would ultimately become the CMG array. Their willingness to let me watch the machining and understand what it takes to turn drawings into parts was as valuable as understanding the physics. These self-described 'Rednecks with Drill Presses' were critical in turning my ideas into reality.

I'd also like to thank the ENY lab technicians working for Jay Anderson. If I had an electrical engineering question, I knew I could ask Sean Miller, if I needed tools and I could talk to Wilber Lacy or Jon Hixenbaugh. Finally, if I needed to buy something Chris Zickefoose was there handle the paperwork usually asking who was paying for it. In short, if I needed it the lab techs were alway there to help.

Finally, I'd like to thank my friends and family. Without you I would have burned out well before accomplishing anything. Thank you for reminding me that there's more to life than lab.

Christopher G. McChesney

# *Table of Contents*

	Page
Abstract . . . . .	iii
Acknowledgements . . . . .	iv
List of Figures . . . . .	vi
List of Tables . . . . .	vii
List of Abbreviations . . . . .	viii
 I. Introduction . . . . .	 1
1.1 Background . . . . .	1
1.2 Problem Statement . . . . .	3
1.3 Research Objectives . . . . .	5
1.4 Methodology . . . . .	5
1.5 Assumptions . . . . .	6
1.6 Preview . . . . .	6
 II. Background . . . . .	 7
2.1 Satellite Simulators . . . . .	7
2.1.1 Micro-Gravity Experiments . . . . .	7
2.1.2 Planar Air-Bearing . . . . .	9
2.1.3 Spherical Air-Bearings . . . . .	9
2.2 AFIT Satellite Simulators . . . . .	12
2.3 Spacecraft Dynamics . . . . .	13
2.3.1 Kinematics . . . . .	14
2.3.2 Rigid Body Dynamics . . . . .	17
2.3.3 Angular Momentum Exchange . . . . .	20
2.3.4 Reaction Wheels . . . . .	22
2.3.5 Control Moment Gyroscopes . . . . .	24
2.4 CMG Momentum Envelope and Steering Laws . . . . .	33
2.4.1 Null Motion . . . . .	36
2.4.2 Moore-Penrose Pseudoinverse Steering Law . . . . .	37
2.4.3 Generalized Inverse Steering Law . . . . .	37
2.5 Linearized Proportional-Integral-Derivative Attitude Control . . . . .	38
2.5.1 PID Control . . . . .	39
2.5.2 Linearized Attitude Control . . . . .	40
2.6 Summary . . . . .	42

	Page
III. Methodology . . . . .	43
3.1 Introduction . . . . .	43
3.2 SimSat System Baseline Hardware . . . . .	43
3.2.1 Ground Station and Data Link . . . . .	43
3.2.2 Air-Bearing . . . . .	44
3.2.3 SimSat Vehicle Hardware . . . . .	45
3.3 SimSat Software Applications . . . . .	48
3.3.1 EPOS User Interface® . . . . .	48
3.3.2 CANOpen Communications Protocol . . . . .	49
3.3.3 SIMULINK® . . . . .	52
3.3.4 dSPACE ControlDesk® . . . . .	54
3.3.5 MATLAB® with dSPACE mLib . . . . .	54
3.4 Reaction Wheel Actuator Changes . . . . .	55
3.4.1 20cm Reaction Wheel . . . . .	55
3.4.2 Electrical Subsystem Protection . . . . .	57
3.5 CMG Development and Sizing . . . . .	58
3.5.1 CMG Array Requirements . . . . .	59
3.5.2 CMG Number and Array Configuration . . . . .	60
3.5.3 CMG Rotor Sizing . . . . .	61
3.5.4 Gimbal Assembly Design . . . . .	63
3.5.5 Gimbal Motor Sizing . . . . .	64
3.5.6 Optical Shaft Encoder . . . . .	65
3.5.7 Gimbal Support and Vibration Isolation . . . . .	66
3.5.8 Gimbal Angle Alignment . . . . .	67
3.6 SimSat Control Program . . . . .	67
3.6.1 Data Collection from Sensors . . . . .	68
3.6.2 Update State Information . . . . .	72
3.6.3 Compute Control Solution . . . . .	73
3.6.4 Generate Actuator Commands . . . . .	74
3.6.5 Transmit Actuator Commands . . . . .	78
3.7 Vehicle Diagnostics . . . . .	78
3.7.1 Vehicle Balancing . . . . .	78
3.7.2 Reaction Wheel Calibration and Tuning . . . . .	80
3.7.3 Vehicle MOI Testing . . . . .	81
3.7.4 CMG Rotor and Gimbal Testing . . . . .	84
3.7.5 CMG and Reaction Wheel Gain Tuning . . . . .	85
3.7.6 CMG and Reaction Wheel Rest-to-Rest Testing . . . . .	86
3.7.7 CMG and Reaction Wheel Torque Testing . . . . .	87
3.7.8 CMG Null Motion . . . . .	88
3.7.9 CMG Torque Multiplication Testing . . . . .	90
3.8 Summary . . . . .	92

	Page
IV. Results and Analysis . . . . .	93
4.1 Introduction . . . . .	93
4.2 Reaction Wheel Calibration and Tuning Results . . . . .	93
4.3 Vehicle MOI Test Results . . . . .	96
4.4 CMG Gimbal Test Results . . . . .	97
4.5 CMG and Reaction Wheel Rest-to-Rest Test Results . . . . .	99
4.5.1 Reaction Wheel Rest-to-Rest Test Results . . . . .	101
4.5.2 CMG Rest-to-Rest Test Results . . . . .	106
4.6 CMG and Reaction Wheel Torque Test Results . . . . .	116
4.6.1 Reaction Wheel Torque Test Results . . . . .	117
4.6.2 CMG Torque Test Results . . . . .	119
4.7 CMG Null Motion Test Results . . . . .	132
4.8 CMG Torque Multiplication Test Results . . . . .	135
4.9 Summary . . . . .	137
V. Conclusions and Recommendations . . . . .	140
5.1 Conclusions . . . . .	140
5.2 Recommendations for Future Development . . . . .	142
5.2.1 Vehicle Hardware Improvements . . . . .	142
5.2.2 Vehicle Software Improvements . . . . .	145
5.2.3 Research Areas . . . . .	146
Appendices . . . . .	148
Appendix A. Supplemental Results Figures . . . . .	149
1.1 Reaction Wheel Calibration Results . . . . .	149
1.2 Vehicle MOI . . . . .	149
1.3 CMG Gimbal Calibration . . . . .	149
1.4 CMG and Reaction Wheel Rest-to-Rest Test Results . . . . .	150
1.4.1 Reaction Wheel Rest-to-Rest Test Results . . . . .	150
1.4.2 CMG Rest-to-Rest Test Results . . . . .	151
1.5 CMG and Reaction Wheel Torque Test Results . . . . .	155
1.5.1 Reaction Wheel Torque Test Results . . . . .	155
1.5.2 CMG Torque Test Results . . . . .	157
1.6 CMG Null Motion Test Results . . . . .	162
1.7 CMG Torque Multiplication Test Results . . . . .	162
Appendix B. SimSat Operating Manual . . . . .	164
2.1 RTICAN Blockset in Simulink . . . . .	164
2.2 SIMULINK Tips . . . . .	166
2.3 mLib Tips . . . . .	168
Bibliography . . . . .	171

## *List of Figures*

Figure		Page
1.1	SimSat II Current Configuration . . . . .	3
2.1	Spherical Air-Bearing Constraints . . . . .	10
2.2	Dumbbell Satellite Simulator . . . . .	10
2.3	Center of Mass - Center of Rotation Misalignment . . . . .	11
2.4	Tabletop Satellite Simulator . . . . .	11
2.5	Umbrella Satellite Simulator . . . . .	12
2.6	SimSat I . . . . .	12
2.7	SimSat II Original Configuration . . . . .	13
2.8	SimSat II with Reaction Wheels . . . . .	14
2.9	Euler Axis $\hat{e}$ - Euler Angle $\phi$ Relating Frame $\{\alpha\}$ and Frame $\{\beta\}$ . . . . .	15
2.10	$\sin\left(\frac{\phi}{2}\right)$ vs. $\left(\frac{\phi}{2}\right)$ . . . . .	17
2.11	SimSat Reaction Wheel . . . . .	22
2.12	CMG Types . . . . .	25
2.13	Four CMG Pyramid Configuration . . . . .	27
2.14	$j^{\text{th}}$ Rotor Frame $\{R_j\}$ . . . . .	28
2.15	$j^{\text{th}}$ Gimbal Frame $\{G_j\}$ . . . . .	29
2.16	$j^{\text{th}}$ CMG Frame $\{C_j\}$ . . . . .	29
2.17	Body Frame $\{b\}$ . . . . .	30
2.18	Angular Momentum Envelope for a Single CMG . . . . .	34
2.19	Angular Momentum Envelopes for a 2 CMGs Array . . . . .	35
2.20	Angular Momentum Envelopes for a 4 CMG Array . . . . .	36
2.21	PID Controller . . . . .	39
2.22	Feedback Linearization . . . . .	41
2.23	Three-Axis Linearized PID Controller . . . . .	42
3.1	SimSat Groundstation . . . . .	44
3.2	Space Electronics, Inc. Tri-Axial Spherical Air-Bearing . . . . .	45
3.3	Mini-Box PC . . . . .	45
3.4	dSPACE MicroAutoBox . . . . .	46
3.5	Northrop Grumman LN-200 Fiber Optic Gyroscope IMU . . . . .	47
3.6	Maxon EPOS Motor Controller . . . . .	48
3.7	EPOS UI Velocity Auto-Tuning Program . . . . .	49

	Page
3.8	CANOpen SDO Protocol . . . . . 50
3.9	CANOpen PDO Protocol . . . . . 51
3.10	SimSat Root-Level SIMULINK Model . . . . . 53
3.11	Example dSPACE ControlDesk Layout . . . . . 54
3.12	10 cm and 20 cm Reaction Wheels . . . . . 56
3.13	Maxon Digital Shunt Regulator 70/30 Block Diagram . . . . . 58
3.14	Four CMG Pyramid Configuration . . . . . 61
3.15	CMG Rotor . . . . . 61
3.16	Gimbal Assembly . . . . . 63
3.17	Gimbal Assembly Balancing . . . . . 64
3.18	CMG Gimbal Motors . . . . . 65
3.19	CMG Gimbal Encoder . . . . . 66
3.20	Gimbal Support Structure . . . . . 66
3.21	Vibration Isolation . . . . . 67
3.22	Gimbal Angle Alignment . . . . . 68
3.23	SIMULINK Model Root . . . . . 69
3.24	CMG Receive Subroutines . . . . . 71
3.25	Vehicle Orientation Update . . . . . 73
3.26	CMG Array Angular Momentum Update . . . . . 74
3.27	Linearized PID Controller . . . . . 75
3.28	GISL Subroutine . . . . . 77
3.29	SimSat Body Axes and Counterweights . . . . . 79
3.30	CMG Rotor Calibration . . . . . 84
3.31	Disturbance Torque Testing . . . . . 89
4.1	X-axis Reaction Wheel Calibration Response -0.25 N-m Commanded Torque . . . . . 95
4.2	Z-axis MOI Test Run . . . . . 96
4.3	X-axis MOI Tests Results . . . . . 98
4.4	Y-axis MOI Tests Results . . . . . 98
4.5	Z-axis MOI Tests Results . . . . . 98
4.6	Gimbal 1 Verification . . . . . 100
4.7	Reaction Wheel Position Hold . . . . . 102
4.8	Reaction Wheel 10° X-axis Slew . . . . . 104
4.9	Reaction Wheel 30° Z-axis Slew . . . . . 105

4.10	CMG Position Hold, MPPSL, $\bar{\delta}_0=[0,0,0,0]$ . . . . .	107
4.11	CMG 10° X-axis Slew, MPPSL, $\bar{\delta}_0=[0,0,0,0]$ . . . . .	108
4.12	CMG 10° X-axis Slew, MPPSL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ . . . . .	109
4.13	CMG 10° X-axis Slew, GISL, $\bar{\delta}_0=[0,0,0,0]$ . . . . .	110
4.14	CMG 10° X-axis Slew, GISL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ . . . . .	111
4.15	CMG 30° Z-axis Slew, MPPSL, $\bar{\delta}_0=[0,0,0,0]$ . . . . .	112
4.16	CMG 30° Z-axis Slew, MPPSL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ . . . . .	113
4.17	CMG 30° Z-axis Slew, GISL, $\bar{\delta}_0=[0,0,0,0]$ . . . . .	114
4.18	CMG 30° Z-axis Slew, GISL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ . . . . .	115
4.19	Reaction Wheel Disturbance Control Test with 0 g . . . . .	117
4.20	Reaction Wheel Disturbance Test X-axis with 17.2 g . . . . .	119
4.21	Reaction Wheel Disturbance Test X-axis with 42.6 g . . . . .	119
4.22	CMG Disturbance Control Test with 0 g . . . . .	120
4.23	CMG Disturbance Test, X-axis MPPSL, $\bar{\delta}_0=[0,0,0,0]$ , 17.2 g . . . . .	121
4.24	CMG Disturbance Test, X-axis MPPSL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ , 17.2 g . . . . .	122
4.25	CMG Disturbance Test, X-axis GISL, $\bar{\delta}_0=[0,0,0,0]$ , 17.2 g . . . . .	123
4.26	CMG Disturbance Test, X-axis GISL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ , 17.2 g . . . . .	124
4.27	CMG Disturbance Test, X-axis MPPSL, $\bar{\delta}_0=[0,0,0,0]$ , 42.6 g . . . . .	125
4.28	CMG Disturbance Test, X-axis MPPSL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ , 42.6 g . . . . .	126
4.29	CMG Disturbance Test, X-axis GISL, $\bar{\delta}_0=[0,0,0,0]$ , 42.6 g . . . . .	127
4.30	CMG Disturbance Test, X-axis GISL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ , 42.6 g . . . . .	128
4.31	CMG Disturbance Test, Z-axis MPPSL, $\bar{\delta}_0=[0,0,0,0]$ , 17.2 g . . . . .	129
4.32	CMG Disturbance Test, Z-axis MPPSL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ , 17.2 g . . . . .	130
4.33	CMG Disturbance Test, Z-axis GISL, $\bar{\delta}_0=[0,0,0,0]$ , 17.2 g . . . . .	131
4.34	CMG Disturbance Test, Z-axis GISL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ , 17.2 g . . . . .	132
4.35	Null Motion Test . . . . .	134
4.36	Gimbal Torque Effect, Rotors Disabled . . . . .	135
4.37	CMG Torque Effect, Rotors Enabled . . . . .	139
A.1	Reaction Wheel Calibration Response -0.25 N-m Commanded Torque . . .	149
A.2	Gimbal Motor Calibration Tests . . . . .	149
A.3	Reaction Wheel 10° Y-axis Slew . . . . .	150
A.4	Reaction Wheel -10° X-axis Slew . . . . .	150
A.5	Reaction Wheel -10° Y-axis Slew . . . . .	150
A.6	Reaction Wheel -30° Z-axis Slew . . . . .	150



A.7	CMG 10° Y-axis Slew, MPPSL, $\bar{\delta}_0=[0,0,0,0]$ . . . . .	151
A.8	CMG 10° Y-axis Slew, MPPSL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ . . . . .	151
A.9	CMG 10° Y-axis Slew, GISL, $\bar{\delta}_0=[0,0,0,0]$ . . . . .	151
A.10	CMG 10° Y-axis Slew, GISL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ . . . . .	151
A.11	CMG -10° X-axis Slew, MPPSL, $\bar{\delta}_0=[0,0,0,0]$ . . . . .	152
A.12	CMG -10° X-axis Slew, MPPSL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ . . . . .	152
A.13	CMG -10° X-axis Slew, GISL, $\bar{\delta}_0=[0,0,0,0]$ . . . . .	152
A.14	CMG -10° X-axis Slew, GISL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ . . . . .	152
A.15	CMG -10° Y-axis Slew, MPPSL, $\bar{\delta}_0=[0,0,0,0]$ . . . . .	153
A.16	CMG -10° Y-axis Slew, MPPSL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ . . . . .	153
A.17	CMG -10° Y-axis Slew, GISL, $\bar{\delta}_0=[0,0,0,0]$ . . . . .	153
A.18	CMG -10° Y-axis Slew, GISL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ . . . . .	153
A.19	CMG -30° Z-axis Slew, MPPSL, $\bar{\delta}_0=[0,0,0,0]$ . . . . .	154
A.20	CMG -30° Z-axis Slew, MPPSL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ . . . . .	154
A.21	CMG -30° Z-axis Slew, GISL, $\bar{\delta}_0=[0,0,0,0]$ . . . . .	154
A.22	CMG -30° Z-axis Slew, GISL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ . . . . .	154
A.23	Reaction Wheel Disturbance Test X-axis with 33.8 g . . . . .	155
A.24	Reaction Wheel Disturbance Test Y-axis with 17.2 g . . . . .	155
A.25	Reaction Wheel Disturbance Test Y-axis with 33.8 g . . . . .	155
A.26	Reaction Wheel Disturbance Test Y-axis with 42.6 g . . . . .	155
A.27	Reaction Wheel Disturbance Test Z-axis with 17.2 g . . . . .	156
A.28	Reaction Wheel Disturbance Test Z-axis with 33.8 g . . . . .	156
A.29	Reaction Wheel Disturbance Test Z-axis with 42.6 g . . . . .	156
A.30	CMG Disturbance Test, X-axis MPPSL, $\bar{\delta}_0=[0,0,0,0]$ , 33.8 g . . . . .	157
A.31	CMG Disturbance Test, X-axis MPPSL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ , 33.8 g . . . . .	157
A.32	CMG Disturbance Test, X-axis GISL, $\bar{\delta}_0=[0,0,0,0]$ , 33.8 g . . . . .	157
A.33	CMG Disturbance Test, X-axis GISL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ , 33.8 g . . . . .	157
A.34	CMG Disturbance Test, Z-axis MPPSL, $\bar{\delta}_0=[0,0,0,0]$ , 33.8 g . . . . .	158
A.35	CMG Disturbance Test, Z-axis MPPSL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ , 33.8 g . . . . .	158
A.36	CMG Disturbance Test, Z-axis GISL, $\bar{\delta}_0=[0,0,0,0]$ , 33.8 g . . . . .	158
A.37	CMG Disturbance Test, Z-axis GISL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ , 33.8 g . . . . .	158
A.38	CMG Disturbance Test, Y-axis MPPSL, $\bar{\delta}_0=[0,0,0,0]$ , 17.2 g . . . . .	158
A.39	CMG Disturbance Test, Y-axis MPPSL, $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ , 17.2 g . . . . .	159
A.40	CMG Disturbance Test, Y-axis GISL, $\bar{\delta}_0=[0,0,0,0]$ , 17.2 g . . . . .	159

	Page
A.41	CMG Disturbance Test, Y-axis GISL, $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 17.2 g . . . . . 159
A.42	CMG Disturbance Test, Y-axis MPPSL, $\bar{\delta}_0=[0,0,0,0]$ , 33.8 g . . . . . 159
A.43	CMG Disturbance Test, Y-axis MPPSL, $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 33.8 g . . . . . 159
A.44	CMG Disturbance Test, Y-axis GISL, $\bar{\delta}_0=[0,0,0,0]$ , 33.8 g . . . . . 160
A.45	CMG Disturbance Test, Y-axis GISL, $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 33.8 g . . . . . 160
A.46	CMG Disturbance Test, Y-axis MPPSL, $\bar{\delta}_0=[0,0,0,0]$ , 42.6 g . . . . . 160
A.47	CMG Disturbance Test, Y-axis MPPSL, $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 42.6 g . . . . . 160
A.48	CMG Disturbance Test, Y-axis GISL, $\bar{\delta}_0=[0,0,0,0]$ , 42.6 g . . . . . 160
A.49	CMG Disturbance Test, Y-axis GISL, $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 42.6 g . . . . . 161
A.50	Null Motion Reaction Wheel Measurements . . . . . 162
A.51	Torque Multiplication Reaction Wheel Measurements . . . . . 163
B.1	RTICAN Write Block . . . . . 165
B.2	RTICAN Request Message Block . . . . . 166
B.3	Subsystem Parameters . . . . . 168
B.4	Control Desk Variable Selection . . . . . 169

*List of Tables*

Table		Page
2.1	AFIT Satellite Simulator Research .....	14
3.1	Reaction Wheel Characteristics .....	57
3.2	CMG Rotor Characteristics .....	63
3.3	Rest-to-Rest Actuator Configurations .....	87
4.1	MOI Test Results .....	97
4.2	Rest-to-Rest Actuator Configurations .....	101

## *List of Abbreviations*

Abbreviation		Page
ACS	Attitude Control System . . . . .	1
CMG	Control Moment Gyroscope . . . . .	1
AFIT	Air Force Institute of Technology . . . . .	3
CAD	Computer Aided Design . . . . .	5
MPPSL	Moore-Penrose Pseudoinverse Steering Law . . . . .	6
GISL	Generalized Inverse Steering Law . . . . .	6
MOI	Moment of Inertia . . . . .	17
PID	Proportional-Integral-Derivative . . . . .	39
CAN	Controller Area Network . . . . .	46
UI	User Interface . . . . .	48
SDO	Service Data Object . . . . .	50
PDO	Process Data Object . . . . .	50
RTR	Remote Transmission Request . . . . .	51
DSR	Digital Shunt Regulator . . . . .	58
TDMA	Time Division Multiple Access . . . . .	70
DAC	Digital to Analog Converter . . . . .	84

# DESIGN OF ATTITUDE CONTROL ACTUATORS FOR A SIMULATED SPACECRAFT

## I. Introduction

### *1.1 Background*

The proper design of the attitude control system (ACS) is one of the most critical aspects of satellite (also referred to as spacecraft or vehicle throughout this document) development. Attitude control is the process by which a spacecraft determines and manipulates its orientation relative to other objects or inertial space. The attitude control requirements are determined by the spacecraft's mission. As an example, a communications spacecraft must point its antennas toward Earth, its solar panels toward the sun, and its thermal radiators away from the sun. The ACS is the spacecraft subsystem responsible for maintaining and manipulating the spacecraft orientation.

While most early satellites utilized passive ACS designs such as spin stabilization, modern satellites utilize an active ACS. An active ACS monitors the spacecraft's orientation, compares it to a desired orientation, and takes corrective action. The corrective action is to either apply an external torque to the vehicle, most often using thrusters, or apply an internal torque by momentum exchange with internal actuators. The ACS must run continuously, as the satellite is always in the presence of disturbance torques that constantly push the the satellite away from its desired orientation. Momentum exchange is the preferred method of control because it offers high fidelity control and does not consume fuel. The two main types of momentum exchange actuators are reaction wheels and control moment gyroscopes (CMGs).

A reaction wheel consists of a flywheel (also referred to as a rotor) and an electric motor. The flywheel's axis of rotation is fixed relative to the spacecraft body. Torque is applied to the flywheel by the motor, altering its rotational speed, and thus its angular momentum, which in turn changes the angular velocity of the spacecraft to maintain the total angular momentum of the vehicle. Three or more reaction wheels, properly

configured, allow for 3-axis control of the spacecraft. Reaction wheels are commonly used on spacecraft as they provide more accurate control than thrusters and do not require fuel.

CMGs, like reaction wheels, utilize the principal of momentum exchange as a means to control the attitude of a spacecraft. A CMG consists of a flywheel mounted on one or more gimbals to change the orientation of the flywheel's spin axis relative to the spacecraft body. The most common CMG designs consist of a flywheel that maintains a constant rotational rate, storing a fixed amount of angular momentum. Torque is applied to the gimbals to change the orientation of the flywheel's angular momentum relative to the spacecraft body. Like the reaction wheel, this change in angular momentum causes an equal but opposite change in the angular momentum of the remainder of the spacecraft. Multiple CMGs are arranged into a CMG array so that total angular momentum of the array can be manipulated as needed by the satellite's control system.

Reaction wheels and CMGs operate on similar principals, but have differing advantages and disadvantages. Reaction wheels, being fixed relative to the body, have simple dynamics. The change in angular momentum determined by the ACS control law is broken into vector components aligned with each reaction wheel making them simple to control. The torque a reaction wheel exerts on the vehicle is equal and opposite of the torque applied by the reaction wheel motor on the flywheel. In order to generate large torques, a large motor with an associated high power draw is required which is its major limitation.

The major advantage of CMGs is their ability to provide torque multiplication. The torque a CMG exerts on the vehicle is significantly larger than the torque input required to rotate the gimbal. Thus, for a given size, weight, and power, a CMG array will produce more torque than a similarly sized reaction wheel array. CMGs are especially useful for large spacecraft, such as the International Space Station, or highly maneuverable satellites like WorldView II.[4] The major drawback to CMGs is that most CMG configurations exhibit complex dynamics and have internal singularities, where control authority disappears before reaching saturation. The cost of addressing the complex dy-

namics has limited the application of CMGs to only those spacecraft with requirements that cannot be met using reaction wheels.

Increases in on-board computer processing capabilities available on spacecraft, as well as the desire to have more maneuverable satellites, have led to a renewed interest in CMGs. Current research is focused on CMG steering algorithm development, how to rotate the CMG gimbals to generate the desired control torques. The final, and arguably most critical phases of algorithm development is verification and validation through hardware-in-the-loop experiments. The expense and risk of just getting a satellite into orbit precludes using untested algorithms on operational spacecraft, especially with systems as critical as the ACS. Satellite simulators offer a way to develop and validate algorithms in a laboratory environment for a fraction of the cost.[27, 26]

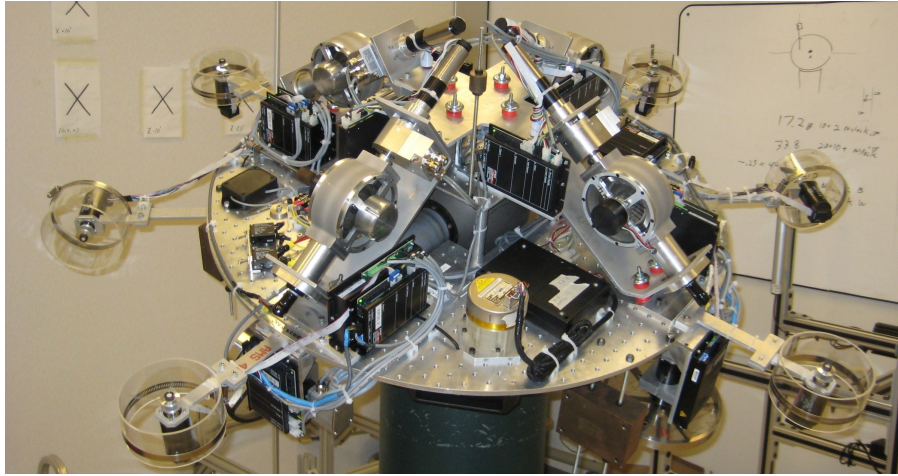


Figure 1.1: SimSat II Current Configuration

## 1.2 Problem Statement

The Air Force Institute of Technology's (AFIT) second-generation satellite simulator SimSat II (hereafter referred to as SimSat), shown in Fig. 1.1, provides AFIT with the ability to conduct attitude control experiments without the risk and cost of a spacecraft launch. As of 2010, SimSat had two sets of actuators, fan-thrusters and reaction wheels.[34] In order to conduct research on CMG algorithms, an affordable CMG array for SimSat was designed to meet the following performance specifications:

1. The CMG array shall generate sufficient control authority for the following capabilities:
  - (a) Positioning accuracy shall be  $\pm 0.01^\circ$ ,
  - (b) A  $\pm 10^\circ$  rest-to-rest maneuver about the  $X$ - and  $Y$ -axes shall be demonstrated within 10 seconds,
  - (c) A  $\pm 30^\circ$  rest-to-rest maneuver about the  $Z$ -axis shall be demonstrated within 10 seconds, and
  - (d) SimSat's angular velocity shall not exceed  $180^\circ/\text{sec}$ .
2. The CMG array torque shall be sufficient to overcome the worst case anticipated disturbance torque, and
3. The CMG array must have sufficient angular momentum storage.
4. The CMG array must interface with SimSat such that
  - (a) Electrical power consumption shall not exceed 20A at 37V,
  - (b) The CMG array must fit within the vehicle, and
  - (c) The CMG array must interface with SimSat's computer system.

Additionally, several improvements were made to the existing reaction wheel subsystem. These improvements included:

1. Resolving actuator communication issues to eliminate timing instability,
2. Increase total angular momentum storage, and
3. Provide physical protection against electrical instability to prevent damage to on-board electronics during large reaction wheel angular velocity changes.

Upgrading SimSat to meet these performance objectives will allow SimSat to support AFIT and Air Force Research Laboratory's research into applying CMG arrays on smaller, more responsive spacecraft.



### ***1.3 Research Objectives***

The primary objectives of this research effort were to design and build a CMG array for AFIT’s SimSat, integrate it with the existing ACS hardware, and validate the performance of the ACS using the CMGs against the requirements in Section 1.2. The secondary objectives were to improve the existing reaction wheel ACS subsystems by addressing known issues with system timing, increasing angular momentum storage, and providing protection against electrical current draw and feedback. The end goal of this research is to provide AFIT with an operational satellite simulator equipped with fan/thruster simulators, reaction wheels, and a 4-unit pyramidal CMG array that can function in concert with one another to provide attitude control.

### ***1.4 Methodology***

The research methodology was designed to address the primary and secondary objectives in tandem whenever possible. Initial research focused on the dynamics, mechanics, and behaviors of various CMG array configurations to determine the tradespace. The CMG array tradespace was systematically narrowed down, ultimately resulting in the selection of the 4-unit pyramidal CMG array. The CMG array high level requirements were developed based on the performance specification in Section 1.2 and extrapolating vehicle growth. These high level requirements were used to scale the component level requirements of the CMGs to select commercial components and design the CMG rotor. The remaining CMG array components were designed around the pyramidal configuration, rotor design, and commercial hardware selected using computer aided design (CAD). CAD was essential because the design requires complex three dimensional geometry, multiple moving assemblies, sub-millimeter tolerances, and conflicting design requirements. Once the CMG array design was frozen, SimSat was disassembled and the existing hardware was relocated to facilitate the installation of the CMG array. During the rebuild process, large diameter reaction wheels were installed to increase the available angular momentum and additional electronics were added to address the electrical feedback issues.

The installation of the CMGs required significant modification to SimSat's existing SIMULINK control program. Rather than extensively modify existing code, a new control program was developed incorporating the best elements of the existing code, addressing the actuator communications and timing issues, and adding the CMG steering laws and actuator control code. The control hardware was then used to validate the functionality of the reaction wheel and CMG hardware, followed by measurement and calibration of SimSat, the reaction wheels, and the CMG array. Lastly, SimSat's ACS was tested against the performance specifications using the reaction wheels and the CMG array. The CMG array was tested using the Moore-Penrose Pseudoinverse Steering Law (MPPSL) and the Generalized Inverse Steering Law (GISL) to compare the behavior of the CMG array with different steering solutions.

### **1.5 Assumptions**

Assumptions are stated where they are made.

### **1.6 Preview**

Chapter II consists of a literature review of topics related to this thesis. These topics include spacecraft dynamics simulators, rigid body dynamics, spacecraft dynamics, and momentum exchange theory. Spacecraft control is also covered in Chapter II. The design, development, construction and testing of the CMG array is covered in Chapter III. Chapter III also covers the modification and testing of the reaction wheel actuators and SimSat's SIMULINK control program. Chapter IV presents the results and analysis of the verification and validation testing performed on the reaction wheels and CMG array. Finally, Chapter V lays out the conclusions of this research and recommendations for future work.

## II. Background

### 2.1 *Satellite Simulators*

Any spacecraft, regardless of size or capability, represents a significant investment of time and capital. As such, testing and validation is, and will always be a critical element in the design process. Because of the importance of testing and validation, most spacecraft use proven, well known subsystems. If a new or unique capability is going to be flown on a spacecraft, it must undergo rigorous testing. The ACS system is no different; however, testing the ACS presents several unique challenges. Computer simulations offer an excellent starting point; however, computer simulations can only model the known behaviors and estimate the known uncertainties, such as disturbance torques. Even the best simulations cannot account for all uncertainties; there will always be unknown uncertainties that impact real systems.

Hardware-in-the-loop experiments offer a way to partially address the problem of unknown uncertainties by testing on physical hardware, with all of the uncertainties that hardware brings. Hardware experiments must also match the relevant aspects of the environment being simulated. There are several aspects to the space environment that can be matched on Earth, such as vacuum, temperature ranges, micro-gravity, and micro-torque, although it is difficult to impossible to match all of them at the same time. For attitude control testing, matching the micro-torque environment is the most crucial of these environmental conditions, because the ACS controls the spacecraft's attitude by applying small torques to the vehicle. The impacts of gravity and friction with any supporting structures produce torques that far exceed the capabilities of most satellite attitude control actuators. These forces must be addressed for accurate testing.[14, 31]

*2.1.1 Micro-Gravity Experiments.* One way to create a micro-torque environment is to also create a micro-gravity environment without going to space, of which there are three main methods currently employed. The first, and most common is to use neutral buoyancy, and is often used by NASA and others for astronaut training. Neutral buoyancy does not lend itself well to attitude control testing, for several reasons. The most important reason is that any fluid dense enough to provide neutral buoyancy to a spacecraft will inherently be too viscous to provide a low torque environment. The re-

sultant drag torques would greatly bias any experiment, and therefore neutral buoyancy is clearly not viable for ACS testing and validation.

The second method of creating a micro-gravity environment without going to space is to conduct drop tower experiments. A drop tower, as its name implies, is an enclosed tower in which experiments are dropped into a net below. During the free fall, the experiment has a few seconds of micro-gravity before it is caught in a net at the bottom. During the micro-gravity phase, the only torques acting on the experiment is air drag, which can be eliminated if the drop tower is partially evacuated. Unfortunately, even the largest drop tower can only provide seconds of micro-gravity, for example NASA Glenn's Zero Gravity Facility can provide only 5.18 seconds of drop time. Worse, at the conclusion of the test hardware must survive rapid deceleration, at NASA Glenn's Zero Gravity Facility the peak acceleration is 65 g. These time constraints and durability requirements limit the applicability of drop towers in the testing of ACS.[22]

The third method of creating micro-gravity environments is by flying an aircraft on zero-g trajectories. NASA's Reduced Gravity Research Program, nicknamed the Vomit Comet, is one example of this type of flight testing. The aircraft flies in alternating parabolic trajectories from 24,000 ft to 34,000 ft in altitude. At the peak of each maneuver, there is a 25 second period of micro-gravity, where the aircraft and everything on-board are in free fall. Unlike a drop tower, the microgravity acts on the air inside the cabin, reducing the effect of air drag on most experiments. After the micro-gravity maneuver, the aircraft experiences approximately 2 g as it pulls out of the dive and back to a climb. Micro-gravity flight testing does address many of the issues associated with the drop tower tests, but the high recurring cost of aircraft operations and the limited duration of tests limit its applicability to ACS testing.[23]

Another option for ACS testing is to accept the effects of gravity and try instead to minimize the net torques and forces acting on the test apparatus. Bearings are mechanical devices that allow motion between two surfaces while minimizing the friction between them. Air-bearings are a specific type of bearing, where the two bearing surfaces are separated by a thin layer of compressed air supplied by one of the bearing surfaces. Because air is a low viscosity fluid, there is nearly zero friction between the two surfaces.

Air-bearings require continuous flow of pressurized air, however several thousand kilograms can be supported with as little as 6 atm of pressure a few liters per minute. The capabilities of air-bearings have made them the preferred testing platform for ground based development ACS related hardware.[31]

*2.1.2 Planar Air-Bearing.* A planar air-bearing consists of a large, flat, smooth plane on which the test apparatus can glide across. Planar air-bearings provide two translational, and one rotational degrees of freedom. An air hockey table is a very simplified example of this, where the puck can move in the horizontal plane, and rotate freely about its vertical axis. Most laboratory planar air-bearings mount air supplies on the test equipment, rather than the bearing surface, but the principal is the same. Planar air-bearings are often used for testing deployment mechanism, robotic actuators, and proximity flight operations. Because planar air-bearings only allow one rotational degree of freedom, they are not typically used for attitude control experiments in which the behavior of all three rotational axis is a concern. With these limitations in mind, planar air-bearings were not examined in detail.

*2.1.3 Spherical Air-Bearings.* A spherical air-bearing consists of a high precision ball and socket joint with the same radius of curvature. The ball and socket are separated by a thin layer of air, minimizing the torque exerted on the sphere. This design provides the spherical air-bearing with three rotational degrees of freedom, and fixes all translational motion. While the spherical air-bearing provides three degrees of freedom, the physical geometry of the socket and pedestal restrict rotation about at least one degree of freedom, as indicated in Fig. 2.1. While there are a wide variety of satellite simulators that use spherical air-bearings, they all fall into three basic designs: dumbbell, tabletop, and umbrella.[31]

The dumbbell design, seen in Fig. 2.2, has the sphere mounted in the center, and each half of the satellite at either end. The dumbbell's unique advantage is that it provide unrestricted movement about two axes, while keeping the center of mass at the center of rotation. If the center of mass and center of rotation are not aligned, then gravity will exert an undesired torque on the vehicle, as shown in Fig. 2.3. AFIT's first satellite

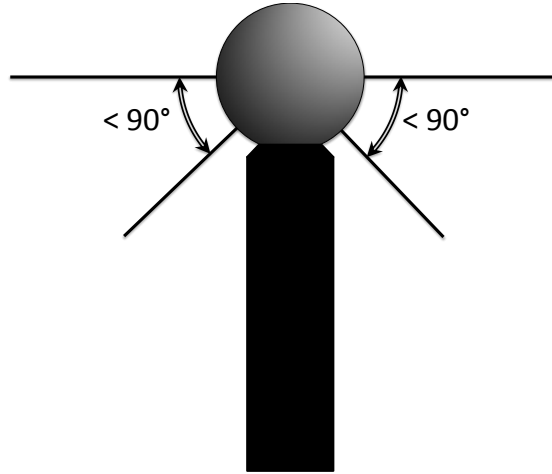


Figure 2.1: Spherical Air-Bearing Constraints

simulator, SimSat I, used this configuration and provided several years of solid research, summarized in Section 2.2. The key disadvantage to the dumbbell design is that the vehicle's geometry inherently results in a large moment of inertia about two axes, which is further discussed in Section 2.3.2.

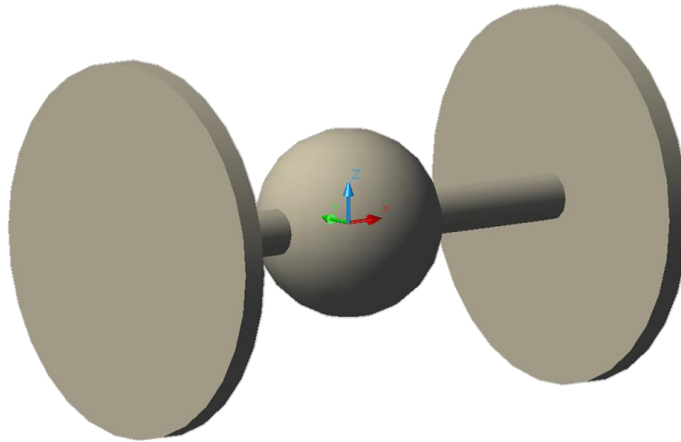


Figure 2.2: Dumbbell Satellite Simulator

The tabletop design, seen in Fig. 2.4, mounts the satellite simulator on the the top half of the air-bearing sphere, and possibly around the outside edge of the lower half of the table top. The inner area on the lower side of the vehicle is kept clear to prevent contact with the pedestal. Like the dumbbell, the tabletop design allows for the center of mass to be placed coincident with the center of rotation, however the tabletop design

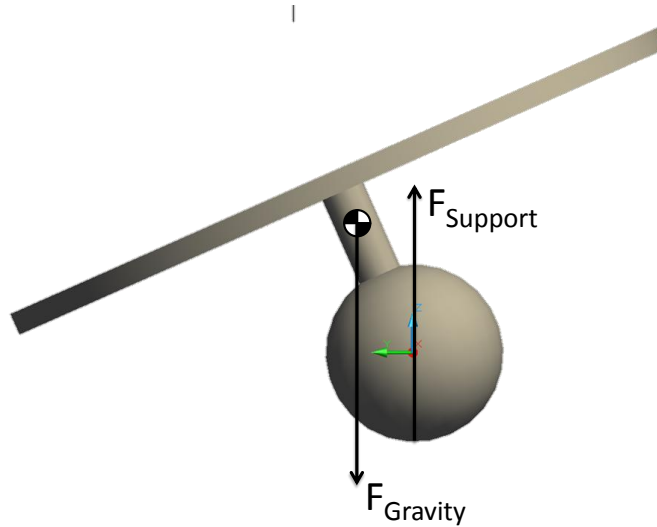


Figure 2.3: Center of Mass - Center of Rotation Misalignment

only has one unrestricted movement axis. AFIT's second generation satellite simulator, SimSat II, is a tabletop style design. While the tabletop design's movement is restricted about two axes, the compact size and small moment of inertia make the design well suited to testing ACS for more maneuverable satellites.

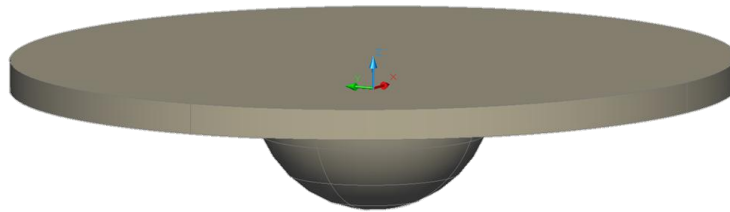


Figure 2.4: Tabletop Satellite Simulator

The umbrella design, seen in Fig. 2.5 is similar to the tabletop design, except a much larger portion of the spherical air-bearing is used. A flat plate can be used, however most large umbrella designs use an umbrella shaped structure to support the experiment hardware. Unlike the tabletop and the dumbbell, the umbrella design is much more difficult to balance, as the physical structure is offset from the air-bearing. Balance can be achieved by use of dense counterweights, and careful design and balancing of the structure. The umbrella configuration, like the tabletop, is still restricted about two axes, but these restrictions are lessened by the geometry.

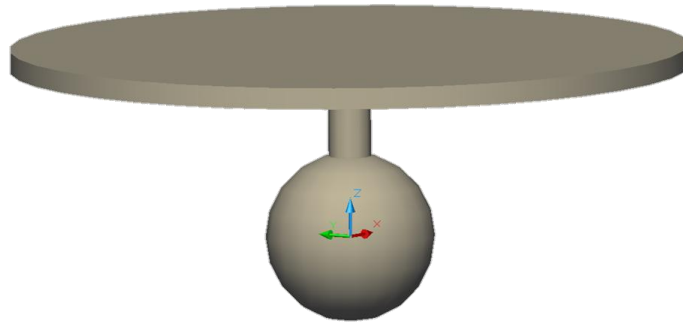


Figure 2.5: Umbrella Satellite Simulator

## 2.2 *AFIT Satellite Simulators*

Since 1999, AFIT has had an satellite simulator to support research and development efforts. Table 2.1 lists the work conducted with SimSat I and SimSat II conducted during that time. The first generation vehicle, SimSat I, was designed by Colebank, Jones, Nagy, Pollak, and Mannebach as the capstone element of their master's degree work.[2] SimSat I, shown in Fig. 2.6 was a dumbbell style satellite simulator with reaction wheels. SimSat I was upgraded in 2003 by French to add improved reaction wheels and cold gas thrusters.[8] Through 2007, SimSat I was used to support research efforts including space situational awareness, autonomous tracking, and fuel estimation. [9, 3, 8, 15, 33, 10, 11] Additionally, SimSat I supported AFIT's spacecraft dynamics courses by providing a hands on demonstrations of attitude dynamics and control.

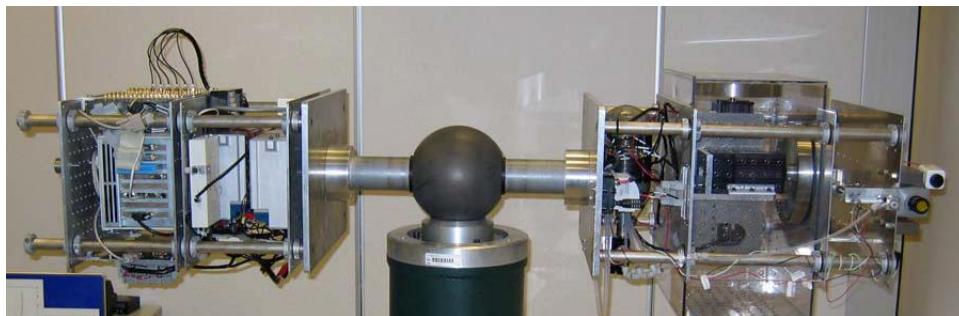


Figure 2.6: SimSat I

While SimSat I provided a solid platform for research, by 2007 it was beginning to show its age. To quote Roach, Rohe, and Welty:



*SIMSAT (I) did have limitations. Due to its large mass and inertia, it was not able to conduct rapid slew maneuvers or to spin up to the required angular velocity necessary to achieve spin stabilization. Also, its momentum wheels quickly saturated while attempting to move the massive simulator. Further, the dumbbell experienced structural deflections that cause its center of mass to move when the simulator rotated about its roll axis. The aging hardware on the SIMSAT was also growing out of date and was in need of replacement.[25]*

Roach *et al.* were commissioned by the faculty to design and build an updated vehicle. The result was SimSat II, shown in Fig 2.7, a tabletop design satellite simulator with up-to-date electronics, a fan/thruster system for control, and provisions for future expansion. In 2008-2009, McFarland conducted the first research on SimSat II, with an investigation of optimal control techniques for attitude control.[21] In 2010, Snider completed SimSat II's first major upgrade, installing a set of three reaction wheels to augment the fan/thruster system. Figure 2.8 shows SimSat II with reaction wheels.

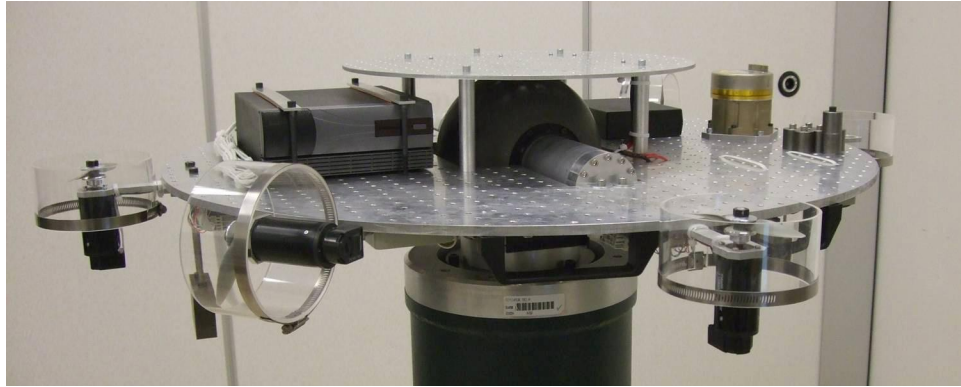


Figure 2.7: SimSat II Original Configuration

Table 2.1: AFIT Satellite Simulator Research

Student(s)	Year	Style	Advisor	Research Topic	Ref
Colebank <i>et al.</i>	1999	Dumbbell	Kramer	Satellite Simulator Design and Assembly	[2]
Fulton	2000	Dumbbell	Agnes	Attitude Control and Multimedia Representation	[9]
Dabrowski	2003	Dumbbell	Cobb	Detection of Parasitic Satellite	[3]
French	2003	Dumbbell	Cobb	Control Strategies for Rapid, Large-Angle Maneuvers	[8]
Kimsal	2004	Dumbbell	Cobb	Autonomous Infrared Tracking	[15]
Smith	2005	Dumbbell	Cobb	Attitude Control using Reaction Wheels and Thrusters	[33]
Geitgey	2006	Dumbbell	Cobb	Measuring Remaining Propellant using Measured MOI	[10]
Hines	2007	Dumbbell	Titus	Fuel Estimation Using Dynamic Response	[11]
Roach <i>et al.</i>	2008	Tabletop	Black	Satellite Simulator Design and Assembly	[25]
McFarland	2009	Tabletop	Swenson	Optimal Control of Spacecraft Reorientation Maneuvers	[21]
Snider	2010	Tabletop	Swenson	Attitude Control of a Satellite Simulator Using Reaction Wheels	[34]

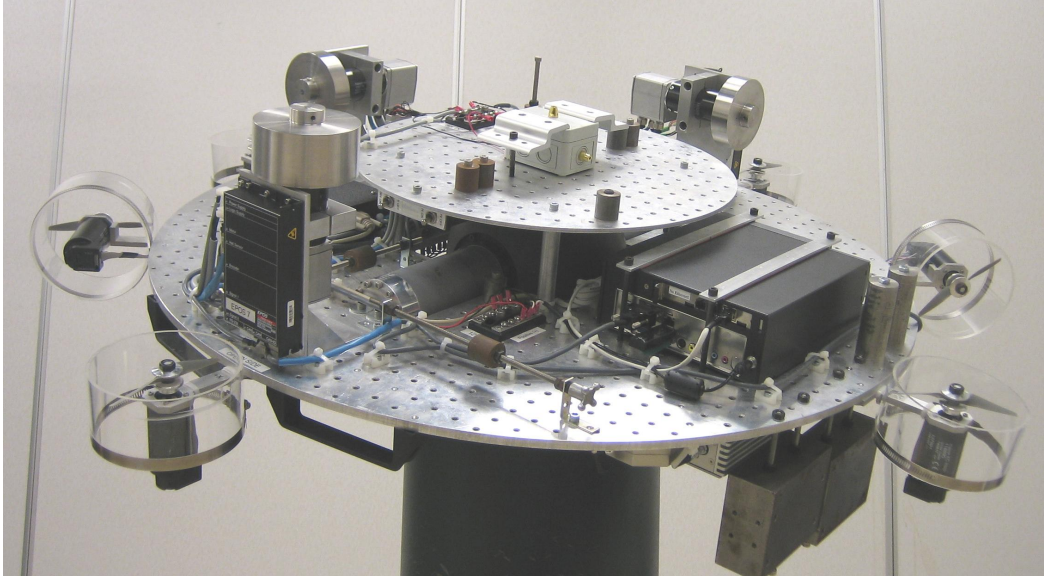


Figure 2.8: SimSat II with Reaction Wheels

### 2.3 *Spacecraft Dynamics*

The purpose of any ACS is to allow the spacecraft to orient itself relative to another object. Examples include orienting antennas toward the Earth, solar arrays toward the sun, and sensors at points of interest. The ACS must be able to manipulate the vehicle's angular position relative to an external references, typically the Earth, sun and/or stars. The starting point for developing an ACS is to derive the satellite's equations of motion, starting with the derivation of the kinematic relationships between reference frames.

*2.3.1 Kinematics.* The first step in controlling angular position, or orientation, is to mathematically define it so that it can be related to other states and controls. There are a number of ways to represent orientation, but the most common for spacecraft are the Euler Parameters, more commonly known as quaternions because of their numerical stability.[12] The quaternion is an extension of Leonhard Euler's 1776 theorem which states:

*In three-dimensional space, any displacement of a rigid body such that a point on the rigid body remains fixed, is equivalent to a single rotation about a fixed axis that runs through the fixed point[7].*

Because reference frames are independent of position, Euler's theorem implies that any two reference frames can be related by a rotation axis defined by a unit vector  $\hat{e}$ , known

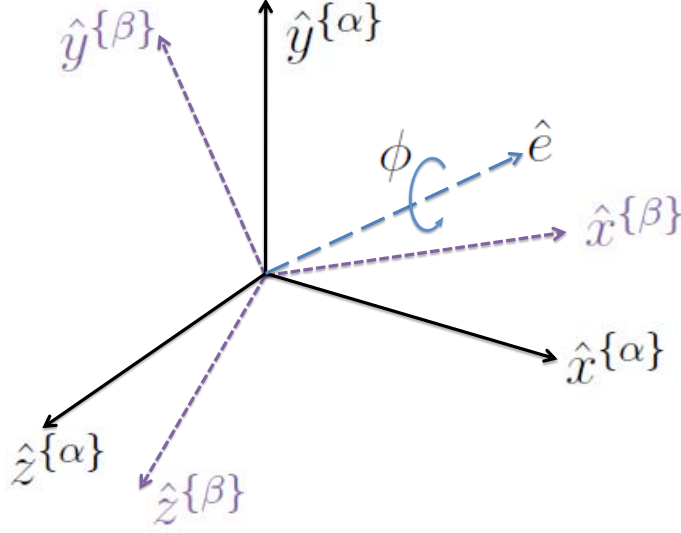


Figure 2.9: Euler Axis  $\hat{e}$  - Euler Angle  $\phi$  Relating Frame  $\{\alpha\}$  and Frame  $\{\beta\}$

as the Euler axis, and an angle  $\phi$ , known as the Euler angle (see Fig. 2.9). The Euler axis/ method has a clear singularity; if  $\phi$  is 0, the Euler axis becomes undefined. The existence of a singularity causes computational problems. One method to remove the singularity is to define a quaternion as

$$\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} e_1 \sin\left(\frac{\phi}{2}\right) \\ e_2 \sin\left(\frac{\phi}{2}\right) \\ e_3 \sin\left(\frac{\phi}{2}\right) \\ \cos\left(\frac{\phi}{2}\right) \end{bmatrix}. \quad (2.1)$$

The terms  $q_1$ ,  $q_2$ , and  $q_3$  are referred to as the quaternion vector  $\vec{q}$ . Using quaternions, the difference between any two coordinate frames or orientations can be described without any singularities, including if the two frames are coincident. The difference between two quaternions is not a simple subtraction; instead it requires the use of a matrix multiplication, resulting in the second orientation being defined relative to the first. For example, given the orientations defined by quaternions  $\bar{a}$  and  $\bar{b}$  that are both defined in

frame  $\{i\}$ , the difference between  $\bar{a}$  and  $\bar{b}$ , defined relative in frame  $\bar{a}$  is

$$\bar{b}^{\{a\}} = \begin{bmatrix} a_4 & a_3 & -a_2 & -a_1 \\ -a_3 & a_4 & a_1 & -a_2 \\ a_2 & -a_1 & a_4 & -a_3 \\ a_1 & a_2 & a_3 & a_4 \end{bmatrix}^T \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}. \quad (2.2)$$

If  $\bar{a}$  and  $\bar{b}$  are the same, then  $\bar{b}^{\{a\}}$  will be  $[0 \ 0 \ 0 \ 1]^T$ . Equation (2.2) is used by the control algorithm to calculate the difference between vehicle's desired orientation and actual orientation to determine the orientation error. The orientation error is a quaternion, and is therefore nonlinear with respect to  $\phi$  which is a potential problem for linear controller. The most straightforward option is to linearize the quaternion about  $\phi = 0$ , thus

$$\begin{bmatrix} e_1 \sin\left(\frac{\phi}{2}\right) \\ e_2 \sin\left(\frac{\phi}{2}\right) \\ e_3 \sin\left(\frac{\phi}{2}\right) \\ \cos\left(\frac{\phi}{2}\right) \end{bmatrix} \approx \begin{bmatrix} e_1 \left(\frac{\phi}{2}\right) \\ e_2 \left(\frac{\phi}{2}\right) \\ e_3 \left(\frac{\phi}{2}\right) \\ 1 \end{bmatrix} \quad (2.3)$$

which is valid for a range of approximately -0.6 to 0.6 radians, as shown in Fig. 2.10. Because the orientation error is calculated relative to the current orientation, the linear approximation is valid for errors less than  $\pm 0.6$  rads, or  $\pm 35^\circ$ , and is sufficient for the tests that will be conducted on SimSat. Aligning the spacecraft to the desired orientation is performed by manipulating the quaternions to match the desired orientation which requires a relationship between the spacecraft's rate of change of orientation  $\dot{\bar{q}}$  and instantaneous angular velocity  $\bar{\omega}$  using the kinematic relationship

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_4 & q_3 & -q_2 \\ -q_3 & q_4 & q_1 \\ q_2 & -q_1 & q_4 \\ -q_1 & -q_2 & -q_3 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}. \quad (2.4)$$

Controlling orientation becomes a matter of manipulating the angular rates, therefore the next step is to derive a relationship between angular rates and the satellite's actuators.

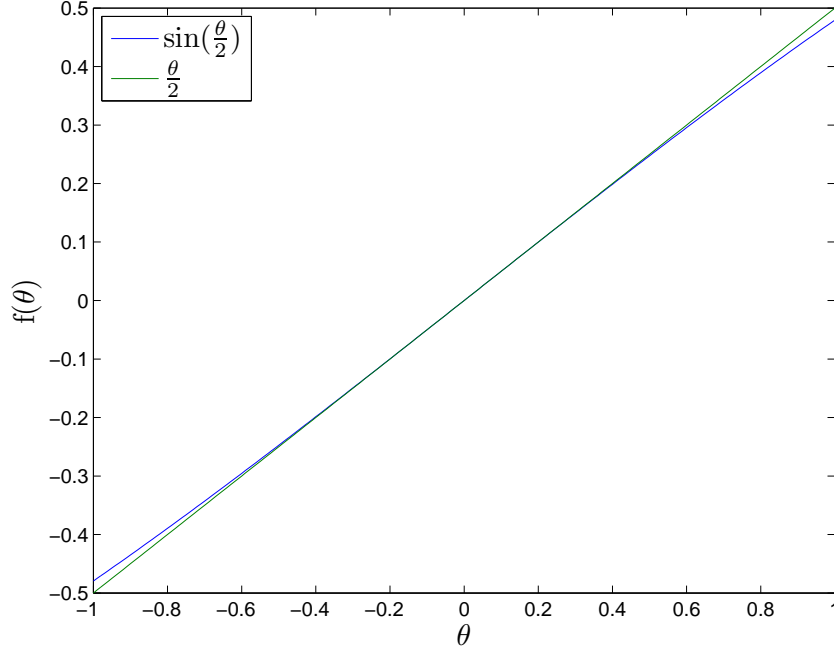


Figure 2.10:  $\sin\left(\frac{\phi}{2}\right)$  vs.  $\left(\frac{\phi}{2}\right)$

*2.3.2 Rigid Body Dynamics.* In order to discuss how angular rates change over time, it is first necessary to discuss the concept of angular momentum. Angular momentum is the rotational analog to linear momentum, defined as

$$\vec{H} = \mathbf{I} \vec{\omega}, \quad (2.5)$$

where  $\mathbf{I}$  is the object's mass moment of inertia (MOI) tensor, and  $\vec{\omega}$  is the object's angular velocity. The MOI is a measure of an object's resistance to changes in angular motion, defined by the object's mass distribution. In Cartesian coordinates, an object's MOI is defined as

$$\mathbf{I} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix} \quad (2.6)$$

where  $I_{xx}$ ,  $I_{yy}$ , and  $I_{zz}$ , known as scalar moments of inertia, are

$$I_{xx} \stackrel{\text{def}}{=} \int_M (y^2 + z^2) dm \quad (2.7a)$$

$$I_{yy} \stackrel{\text{def}}{=} \int_M (x^2 + z^2) dm \quad (2.7b)$$

$$I_{zz} \stackrel{\text{def}}{=} \int_M (x^2 + y^2) dm \quad (2.7c)$$

and the remaining terms, known as products of inertia, are

$$I_{xy} = I_{yx} \stackrel{\text{def}}{=} \int_M (xy) dm \quad (2.8a)$$

$$I_{xz} = I_{zx} \stackrel{\text{def}}{=} \int_M (xz) dm \quad (2.8b)$$

$$I_{yz} = I_{zy} \stackrel{\text{def}}{=} \int_M (yz) dm \quad (2.8c)$$

where  $x$ ,  $y$ , and  $z$  are the distance of the differential mass  $dm$  from the center of rotation, along an arbitrary orthogonal set. From Eqns. (2.7) and (2.8) it is clear that an object's MOI depends on the point of rotation and the reference frame being used. Its convenient to specify a coordinate system with its origin to be the object's center of mass and its directions such that the products of inertia are zero, or

$$I_{xy} = I_{xz} = I_{yz} = 0 \quad (2.9)$$

which maximizes the moments of inertia, which are traditionally called the principal moments of inertia, and are along coordinate axes traditionally called the principal axes. The principal axes define the body reference frame  $\{b\}$ . The MOI of a rigid body about its center of mass, expressed in the body frame, is

$$\mathbf{I}_b = \begin{bmatrix} I_1 & 0 & 0 \\ 0 & I_2 & 0 \\ 0 & 0 & I_3 \end{bmatrix} \quad (2.10)$$

and is constant for a rigid body. For symmetric objects, the axis of symmetry will be a principal axis. Additionally, because  $\mathbf{I}_b$  is diagonal  $\mathbf{I}_b^{-1}$  is easy to compute being

$$\mathbf{I}_b^{-1} = \begin{bmatrix} I_1^{-1} & 0 & 0 \\ 0 & I_2^{-1} & 0 \\ 0 & 0 & I_3^{-1} \end{bmatrix}. \quad (2.11)$$

With angular momentum now defined, we can turn our attention to manipulating angular momentum. Newton's second law, defined for linear momentum, states that

*The time rate of change of linear momentum of a body is equal to the sum of the forces acting on that body.[32]*

The rotational analog to Newton's second law states that the time rate of change of centroidal angular momentum in inertial space is equal to the sum of the moments acting on the body, mathematically

$$\vec{M} = \dot{\vec{H}} = \frac{d}{dt}^{\{i\}} \vec{H} \quad (2.12)$$

where  $\vec{M}$  represents the applied moments and  $\dot{\vec{H}}$  represents time rate of change of centroidal angular momentum with respect to an inertial reference frame  $\{i\}$ . It is important to emphasize that Eq. (2.5), like Newton's second law, is only valid if the derivative is taken in an inertial reference frame. Differentiating Eq. (2.5) in an inertial frame requires the use of the chain rule for derivatives as both  $\mathbf{I}$  and  $\vec{\omega}$  change with respect to time in an inertial frame as the body rotates. Differentiation of Eq. (2.5) using the chain rule results in

$$\frac{d}{dt}^{\{i\}} \vec{H} = \left( \frac{d}{dt}^{\{i\}} \mathbf{I} \right) \vec{\omega} + \mathbf{I} \left( \frac{d}{dt}^{\{i\}} \vec{\omega} \right). \quad (2.13)$$

Equation (2.13) requires that  $\mathbf{I}$  be differentiated in the inertial frame. As shown in Eqns. (2.7) and (2.8), differentiating an object's MOI in any other frame except a body fixed frame is difficult, as the mass distribution changes as the object rotates relative to the coordinate frame. Alternatively, the derivative can be taken in the body frame  $\{b\}$ , where the MOI remains constant, however this requires changing reference frames while accounting for the relative motion of the two frames. Changing reference frames requires

using the transport theorem to compute derivatives in different frames [30]

$$\frac{d}{dt}^{\{i\}} \vec{f} = \frac{d}{dt}^{\{r\}} \vec{f} + \vec{\omega}_{ri} \times \vec{f} \quad (2.14)$$

where  $\vec{f}$  is an arbitrary vector,  $\{r\}$  is an arbitrary reference frame, and  $\vec{\omega}_{ri}$  is the angular velocity of the  $\{r\}$  frame with respect to the  $\{i\}$  frame. The transport theorem maintains the relative motion of two reference frames while allowing differentiation in either frame. The transport theorem can be applied to Eq. (2.12), selecting the body frame  $\{b\}$  where  $\mathbf{I}$  is constant, which is written as

$$\frac{d}{dt}^{\{i\}} \vec{H} = \mathbf{I}_b \frac{d}{dt}^{\{b\}} \vec{\omega}_{bi} + \vec{\omega}_{bi} \times \mathbf{I}_b \vec{\omega}_{bi} \quad (2.15)$$

where  $\mathbf{I}_b$  is the MOI expressed in the  $\{b\}$  frame about the center of mass and  $\vec{\omega}_{bi}$  is the angular rate of both the body and body frame relative to an inertial frame. Substituting this into Eq. (2.12) yields

$$\vec{M} = \mathbf{I}_b \frac{d}{dt}^{\{b\}} \vec{\omega}_{bi} + \vec{\omega}_{bi} \times \mathbf{I}_b \vec{\omega}_{bi}, \quad (2.16)$$

which is commonly known as Euler's equation for rotational bodies, written in vector form. Converting Eq. (2.16) to Newtonian notation results in

$$\vec{M} = \mathbf{I}_b \dot{\vec{\omega}}_{bi} + \vec{\omega}_{bi} \times \mathbf{I}_b \vec{\omega}_{bi} \quad (2.17)$$

where all vectors are expressed in the  $\{b\}$  frame. Euler's equation allows for analysis of the spacecraft dynamics while operating in the body frame. Except where explicitly stated, all equations for the remainder of Section 2.3 are expressed in the body frame  $\{b\}$ .

*2.3.3 Angular Momentum Exchange.* Euler's equations assume the spacecraft is single rigid body, but a spacecraft containing movable actuators clearly is not. In order to apply the equations developed in Section 2.3.2, it is necessary to break the spacecraft's



angular momentum up as

$$\vec{H}_{net} = \vec{h}_{body} + \vec{h}_{act} \quad (2.18)$$

where  $\vec{h}_{act}$  represents the angular momentum of the actuator defined at the actuator's center of mass and  $\vec{h}_{body}$  represents the angular momentum of the vehicle body. If the moving components are symmetric about their axis of rotation, such that their movement does not change the vehicle's MOI, then Eq. (2.18) can be rewritten as

$$\vec{H}_{net} = \mathbf{I}_b \vec{\omega}_{bi} + \vec{h}_{act} \quad (2.19)$$

where  $\mathbf{I}_b$  contains both the vehicle's static MOI and the MOI of each actuator combined using the parallel axis theorem. The actuator's angular momentum  $\vec{h}_{act}$  contains only the dynamic angular momentum of the actuators. Requiring that these moving components be symmetric about their axis of rotation implies an actuator design constraint. If the actuators are not symmetric, movement of the actuators will alter the MOI of the vehicle. In practice, the actuators have significantly smaller MOI values than their host spacecraft, which simplifies the design requirement to having the actuator's center of mass located along the axis of rotation. Substituting Eq. (2.19) into Eq. (2.17) results in

$$\vec{M} = \mathbf{I}_b \dot{\vec{\omega}}_{bi} + \dot{\vec{h}}_{act} + \vec{\omega}_{bi} \times (\mathbf{I}_b \vec{\omega}_{bi} + \vec{h}_{act}). \quad (2.20)$$

Now, if the applied external moments are assumed to be negligible, a valid assumption for most spacecraft over short time spans, Eq. (2.20) can be re-arranged, such that

$$\mathbf{I}_b \dot{\vec{\omega}}_{bi} = -\dot{\vec{h}}_{act} - \vec{\omega}_{bi} \times (\mathbf{I}_b \vec{\omega}_{bi} + \vec{h}_{act}). \quad (2.21)$$

From Eq. (2.21), it is clear that changing  $\dot{\vec{h}}_{act}$  in magnitude or direction will cause a change in  $\dot{\vec{\omega}}_{bi}$ . In effect, angular momentum is exchanged between the actuator and the body, hence their description as 'Momentum Exchange Devices.' This exchange of momentum imparts a torque on the vehicle. The term  $-\vec{\omega}_{bi} \times (\mathbf{I}_b \vec{\omega}_{bi} + \vec{h}_{act})$  creates a nonlinear exchange in angular momentum between the actuator and the body. In many cases, the spacecraft's body rates  $\vec{\omega}_{bi}$  is assumed to be small, and these terms can be

ignored. Alternatively, a nonlinear feedback controller can be used to directly address the nonlinear coupling. The two main types of momentum exchange devices are reaction wheels and CMGs. Reaction wheels will be covered first, followed by CMGs.

*2.3.4 Reaction Wheels.* A reaction wheels consists of a flywheel, electric motor, and supporting electronics. One of SimSat's reaction wheel is shown in Fig. 2.11. The reaction wheel motor is mounted rigidly to the spacecraft body, fixing its axis of rotation in the body frame. The angular momentum of an individual reaction wheel is

$$\vec{h}_i = I_{rw} \vec{\Psi}_i \quad (2.22)$$

where  $I_{rw}$  is the reaction wheel's scalar moment of inertia along its axis of rotation

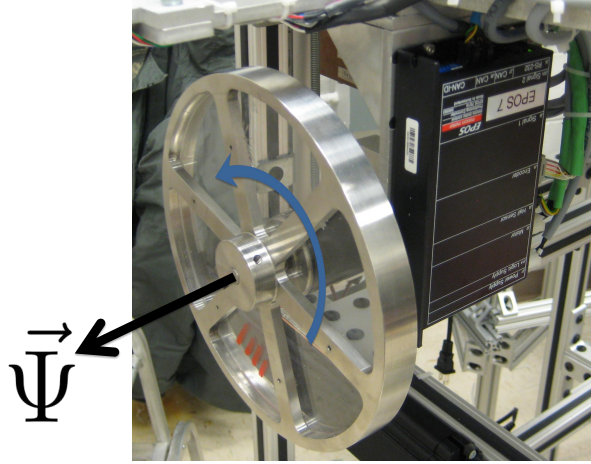


Figure 2.11: SimSat Reaction Wheel

and  $\vec{\Psi}_i$  is the angular rate vector of the reaction wheel defined in the body frame of the spacecraft, shown in Fig 2.11. In order to control the vehicle in all three axes, a minimum of three reaction wheels are required. For this discussion, assume there are only orthogonal three reaction wheels aligned with the spacecraft principal axes. The angular momentum vector  $\vec{\Psi}_1$  is aligned with the body  $x$ -axis,  $\vec{\Psi}_2$  with the body  $y$ -axis, and  $\vec{\Psi}_3$  with the body  $z$ -axis. Reaction wheels are fixed in the body, thus the direction of  $\vec{\Psi}_i$  and the value of  $I_{rw}$  are constant; only the angular rate magnitude  $|\vec{\Psi}_i|$  can be changed. For three orthogonal reaction wheels, the angular momentum of the reaction

wheel array is

$$\vec{h}_{rwa} = \begin{bmatrix} I_{rw} \Psi_1 \\ I_{rw} \Psi_2 \\ I_{rw} \Psi_3 \end{bmatrix}. \quad (2.23)$$

Substituting  $\vec{h}_{rwa}$  in for  $\vec{h}_{act}$  in the previous model, Eqns. (2.19) and (2.20), yields

$$\vec{H}_{net} = \mathbf{I}_b \vec{\omega}_{bi} + \begin{bmatrix} I_{rw} \Psi_1 \\ I_{rw} \Psi_2 \\ I_{rw} \Psi_3 \end{bmatrix}, \quad (2.24)$$

and

$$\vec{M} = \mathbf{I}_b \dot{\vec{\omega}}_{bi} + \begin{bmatrix} I_{rw} \dot{\Psi}_1 \\ I_{rw} \dot{\Psi}_2 \\ I_{rw} \dot{\Psi}_3 \end{bmatrix} + \vec{\omega}_{bi} \times \left( \mathbf{I}_b \vec{\omega}_{bi} + \begin{bmatrix} I_{rw} \Psi_1 \\ I_{rw} \Psi_2 \\ I_{rw} \Psi_3 \end{bmatrix} \right), \quad (2.25)$$

respectively, where  $\dot{\Psi}_i$  is each reaction wheel's acceleration. At this point, it is important to note that

$$M_{rw} = I_{rw} \dot{\Psi}_{rw} \quad (2.26)$$

where  $M_{rw}$  is the torque applied by the reaction wheel motor and  $\dot{\Psi}_{rw}$  is the reaction wheel acceleration. This means that the torque applied to the vehicle is equal and opposite of the torque applied to the reaction wheel. Applying the earlier assumption that the external moments can be neglected, Eq. (2.25) can be re-written as

$$\mathbf{I}_b \dot{\vec{\omega}}_{bi} = - \begin{bmatrix} I_{rw} \dot{\Psi}_1 \\ I_{rw} \dot{\Psi}_2 \\ I_{rw} \dot{\Psi}_3 \end{bmatrix} - \vec{\omega}_{bi} \times \left( \mathbf{I}_b \vec{\omega}_{bi} + \begin{bmatrix} I_{rw} \Psi_1 \\ I_{rw} \Psi_2 \\ I_{rw} \Psi_3 \end{bmatrix} \right). \quad (2.27)$$

The controller specifies changes to the angular momentum of the reaction wheel array by adjusting the angular acceleration  $\dot{\Psi}_i$  in order to change the vehicle's corresponding angular rotational acceleration  $\dot{\omega}_i$ . The additional  $\vec{\omega}_{bi}$  cross products in Eq. (2.27) add nonlinear effects which are addressed in Section 2.5.2.

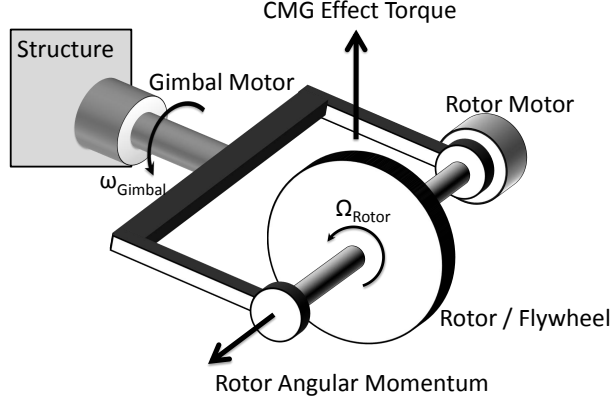
As mentioned previously, reaction wheels have fundamental limitations. The maximum torque that a reaction wheel can impart on the vehicle is determined directly by the torque generated by the motor, as shown in Eq. (2.26). The only way to increase this torque available is to use a larger, and consequently heavier, motor that consumes more power. Additionally, the power consumed by an electric motor is

$$P = \eta_{motor} \tau \Psi \quad (2.28)$$

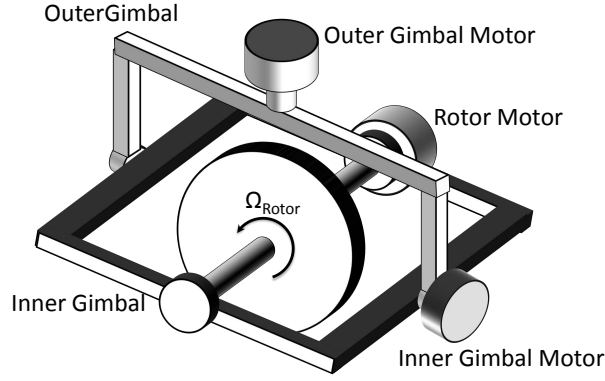
where  $\eta_{motor}$  is the motor's efficiency which is a function of angular rate  $\Psi$ , and  $\tau$  is the applied torque. Equation (2.28) shows that the power required by a reaction wheel increases as the angular momentum stored increases.

Additionally, it should be noted that most spacecraft use four reaction wheels in a pyramidal arrangement so that if any one reaction wheel fails, torques can still be generated in all three directions. Because reaction wheels do not change orientation relative to the body, the formulation of the spacecraft dynamics as described by Eq. (2.27) does not change as the reaction wheels change angular velocity.

*2.3.5 Control Moment Gyroscopes.* The second category of momentum exchange devices is the control moment gyroscope (CMG). A CMG exchanges momentum with the spacecraft by rotating a flywheel mounted on a gimbaled platform. The flywheel is typically spun at a constant rate, and therefore maintains a fixed magnitude of angular momentum. Torque is applied to the gimbals to change the direction of the angular momentum vector, which imparts a torque on the vehicle. There are two main types of CMGs, single gimbal and dual gimbal, shown in Figs. 2.12 (a) and (b), respectively. A single gimbal CMG can produce torque only perpendicular to its gimbal axis, while a dual gimbal CMG can produce torque along two axes. The direction of torque production in either case is dependent on the current angle of the gimbals. Because single gimbal CMGs are being used on SimSat, the remainder of this analysis will focus on their dynamics; the dual gimbal CMG is noted for completeness. Additionally, for the remainder of this document CMG refers specifically to single gimbal CMGs. Figure 2.12(a) shows that the torque generated by the CMG will be perpendicular to both the angular mo-



(a) Single Gimbal CMG



(b) Dual Gimbal CMG

Figure 2.12: CMG Types

mentum vector and the gimbal axis. Neglecting the gimbal inertia, the torque produced is

$$\vec{\tau} = \dot{\vec{h}} = \vec{\omega}_{gimbal} \times \vec{h}_{rotor} \quad (2.29)$$

where  $\vec{\omega}$  is the gimbal rate and  $\vec{h}$  is the angular momentum of the CMG flywheel. Torque multiplication occurs because it requires only a small amount of torque to impart an angular velocity on the gimbal  $\vec{\omega}$  which changes the direction of the angular momentum stored in the rotor and imparts a large CMG effective torque on the spacecraft. The torque multiplication effect means the torque capabilities of a CMG array are not limited by the power of the gimbal motors. Torque multiplication makes CMG arrays useful in

situations where large torques are required and reaction wheels become impractical due to size, weight, or power limitations. This situation occurs with large spacecraft, such as the International Space Station, Skylab, and Mir, or agile spacecraft like Worldview II [37][4]. This large torque envelope does not come without a cost; CMG arrays have complex dynamics, as will be discussed shortly, and require equally complex control algorithms.

Unlike a reaction wheel, where the torque vector's direction remains constant with respect to the body, the torque produced by a CMG is dependent on the gimbal position, which changes as torque is generated. In order to provide three axis control, multiple CMGs are arranged in an array. The number and orientation of CMGs determines the overall performance of the array. There are several factors to consider when selecting the CMG array design: system dynamics, physical space limitations, costs, and singularity concerns. While a three unit CMG array can provide three axis control, the need for redundancy leads to four unit CMG arrays being the minimum number used on spacecraft. Because the CMG gimbals and rotors rotate relative to the body, the system dynamics are highly dependent on the CMG array configuration. Therefore, the control laws developed for one CMG array design may not be applicable to others. The four unit pyramid design was chosen for SimSat because it offers a near spherical momentum envelope. This provides the best compromise for high torque in all directions and allows for use of the entire momentum envelope.[16]. Figure 2.13 shows the standard four unit pyramid configuration, the CMG array design implemented on SimSat.

With the pyramid array selected, the system dynamics can now be derived. Like we did previously with the reaction wheels, the starting point is to separate the angular momentum of the spacecraft into the fixed body component and CMG array components, per Eqns. (2.19) and (2.20), which are restated below for convenience as

$$\vec{H}_{net} = \mathbf{I}_b \vec{\omega}_{bi} + \vec{h}_{act} \quad (2.30)$$

and

$$\vec{M} = \mathbf{I}_b \dot{\vec{\omega}}_{bi} + \dot{\vec{h}}_{act} + \vec{\omega}_{bi} \times (\mathbf{I}_b \vec{\omega}_{bi} + \vec{h}_{act}) \quad (2.31)$$

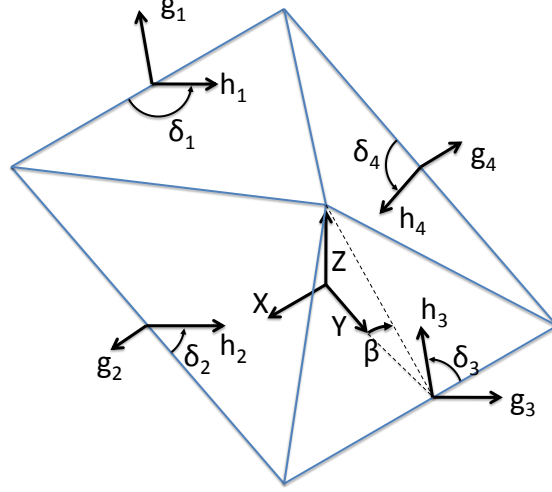


Figure 2.13: Four CMG Pyramid Configuration

where  $\vec{h}_{act}$  represents the total angular momentum of the CMG array. In order to use Eqns. (2.30) and (2.31),  $\vec{h}_{act}$  and  $\dot{\vec{h}}_{act}$  must be expressed in the body frame. The simplest way to accomplish this is to define  $\vec{h}_{act}$  with respect to a reference frame defined by the gimbal assembly and rotate  $\vec{h}_{act}$  back to the body frame using the geometry of the CMG array. The first reference frame is defined by the rotor and gimbal axes, shown in Fig. 2.14, of the  $j^{th}$  CMG, referred to as the rotor reference frame  $\{R_j\}$ . The angular momentum of the rotor and gimbal structure, defined in  $\{R_j\}$  frame, is

$$\vec{h}_j^{\{R_j\}} = \begin{bmatrix} I_G \dot{\delta}_j \\ 0 \\ I_R \Omega \end{bmatrix} \quad (2.32)$$

where  $I_G$  represents the scalar moment of inertia of the gimbal assembly about the gimbal axis,  $\delta_j$  is the instantaneous gimbal angle, and  $\dot{\delta}_j$  gimbal angular rate of the  $j^{th}$  gimbal,  $I_R$  is the scalar moment inertia of the rotor about the rotor axis, and  $\Omega$  is the fixed rotational rate of the rotor. By design the CMGs were constructed to be identical, thus  $I_G$ ,  $I_R$ , and  $\Omega$  should be the same for each CMG. Measurements were taken during the design and construction stages and adjustments were made to ensure that this assumption is valid for the CMGs installed on SimSat.

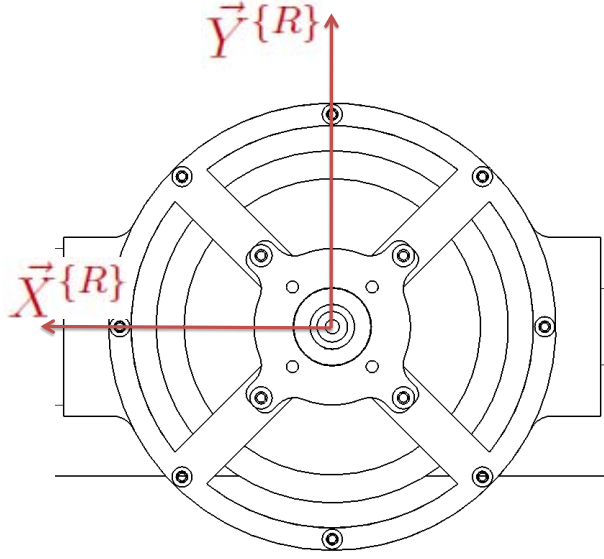


Figure 2.14:  $j^{\text{th}}$  Rotor Frame  $\{R_j\}$

In order to apply the momentum exchange equations derived previously, the rotor and gimbal angular momentum  $\vec{h}_j$  must be expressed in the the body frame  $\{b\}$ . The angular momentum  $\vec{h}_j$  can be transformed from each individual rotor frame  $\{R_j\}$  to the vehicle's body frame  $\{b\}$  through a sequence of rotations (see Figs. 2.15 through 2.17). The rotor and gimbal angular momentum can be expressed in the body frame as

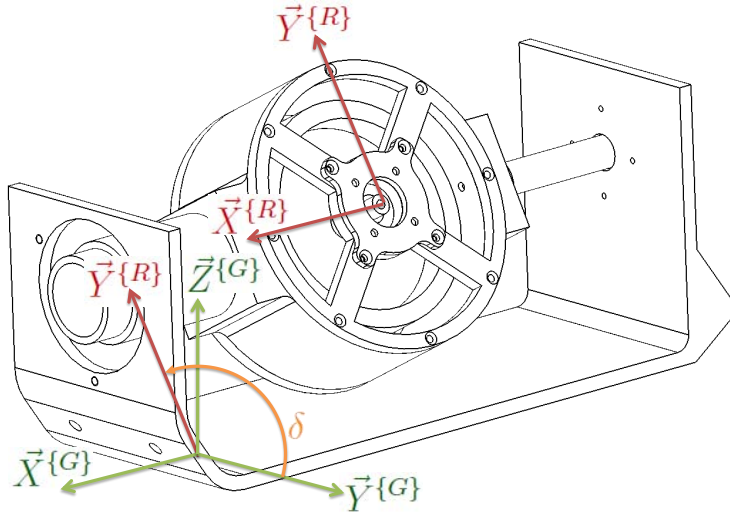


Figure 2.15:  $j^{\text{th}}$  Gimbal Frame  $\{G_j\}$



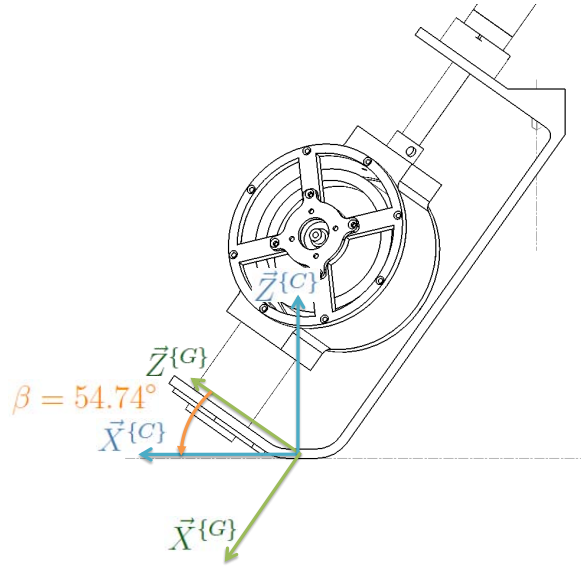


Figure 2.16:  $j^{\text{th}}$  CMG Frame  $\{C_j\}$

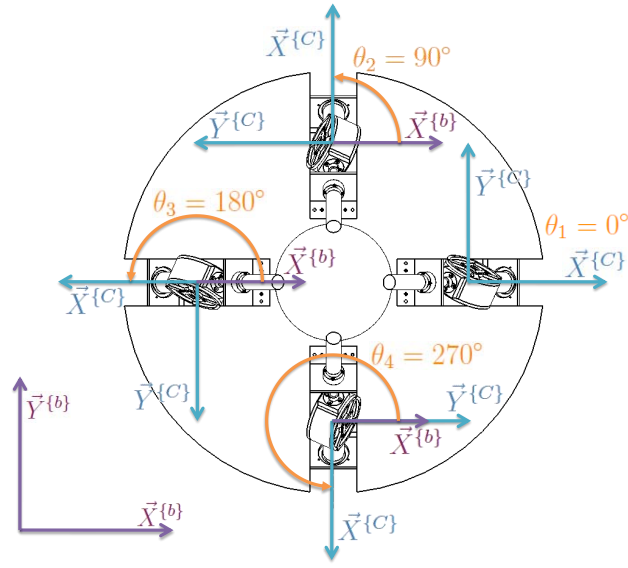


Figure 2.17: Body Frame  $\{b\}$

$$\vec{h}_j^{\{b\}} = \mathbf{R}_3(\theta_j)^T \mathbf{R}_2(\beta)^T \mathbf{R}_1(\delta_j)^T \begin{bmatrix} I_G \dot{\delta}_j \\ 0 \\ I_R \Omega \end{bmatrix}. \quad (2.33)$$

The angle  $\theta_j$  is the mounting angle of the individual CMG frames with respect to the vehicle about the body frame  $z$ -axis  $\{b_3\}$  with each CMG having its own value for  $\theta_j$  (see Fig. 2.17). The angle  $\beta$  is defined as the mounting angle between the CMG gimbal axis

and the body  $XY$ -plane  $\{b_1 - b_2\}$ , shown in Fig. 2.16 and is the same for all CMGs at  $54.74^\circ$ . The mounting angles  $\theta_j$  and  $\beta$  were chosen to provide a near spherical momentum envelope which is addressed in Section 2.4. A spherical momentum envelope ensures that the combined angular momentum of the CMG array can be oriented in any direction relative to the body. With these angle restrictions in mind, Eq. (2.33) can be simplified to

$$\vec{h}_j^{\{b\}} = \mathbf{R}_3(\theta_j)^T \begin{bmatrix} I_G \cos(\beta) \dot{\delta}_j + I_R \Omega \sin(\beta) \cos(\delta_j) \\ -I_R \Omega \sin(\delta_j) \\ -I_G \sin(\beta) \dot{\delta}_j + I_R \Omega \cos(\beta) \cos(\delta_j) \end{bmatrix}. \quad (2.34)$$

The angular momentum of the CMG array is the sum of the four individual CMGs, expressed as

$$\vec{h}_{cmga} = \sum_{j=1}^4 \left( \mathbf{R}_3(\theta_j)^T \begin{bmatrix} I_G \cos(\beta) \dot{\delta}_j + I_R \Omega \sin(\beta) \cos(\delta_j) \\ -I_R \Omega \sin(\delta_j) \\ -I_G \sin(\beta) \dot{\delta}_j + I_R \Omega \cos(\beta) \cos(\delta_j) \end{bmatrix} \right). \quad (2.35)$$

As with the reaction wheel array,  $\vec{h}_{cmga}$  can be substituted for  $\vec{h}_{act}$  in the simplified model Eqs. (2.19), resulting in

$$\vec{H}_{net} = \mathbf{I}_b \text{vec} \omega_{bi} + \sum_{j=1}^4 \left( \mathbf{R}_3(\theta_j)^T \begin{bmatrix} I_G \cos(\beta) \dot{\delta}_j + I_R \Omega \sin(\beta) \cos(\delta_j) \\ -I_R \Omega \sin(\delta_j) \\ -I_G \sin(\beta) \dot{\delta}_j + I_R \Omega \cos(\beta) \cos(\delta_j) \end{bmatrix} \right). \quad (2.36)$$

Differentiating Eq. (2.19) with respect to an inertial frame results in

$$\begin{aligned} \dot{\vec{H}}_{net} = \mathbf{I}_b \dot{\vec{\omega}}_{bi} + \sum_{j=1}^4 \left( \mathbf{R}_3(\theta_j)^T \begin{bmatrix} I_G \cos(\beta) \ddot{\delta}_j - I_R \Omega \sin(\beta) \sin(\delta_j) \dot{\delta}_j \\ -I_R \Omega \cos(\delta_j) \dot{\delta}_j \\ -I_G \sin(\beta) \ddot{\delta}_j - I_R \Omega \cos(\beta) \sin(\delta_j) \dot{\delta}_j \end{bmatrix} \right) + \\ \vec{\omega}_{bi} \times \mathbf{I}_b \vec{\omega}_{bi} + \vec{\omega}_{bi} \times \sum_{j=1}^4 \left( \mathbf{R}_3(\theta_j)^T \begin{bmatrix} I_G \cos(\beta) \dot{\delta}_j + I_R \Omega \sin(\beta) \cos(\delta_j) \\ -I_R \Omega \sin(\delta_j) \\ -I_G \sin(\beta) \dot{\delta}_j + I_R \Omega \cos(\beta) \cos(\delta_j) \end{bmatrix} \right). \end{aligned} \quad (2.37)$$

Equation (2.37) can be simplified if certain assumptions are made, specifically

$$\dot{\delta}_j \ll \Omega \quad (2.38)$$

$$\ddot{\delta}_j \ll \Omega \dot{\delta}_j \quad (2.39)$$

which are valid for CMGs as the rotor rate  $\Omega$  is typically hundreds to thousands of radians per second, while gimbal rates  $\dot{\delta}$  and accelerations  $\ddot{\delta}$  are limited to a few radians per second and radians per second per second. The CMGs installed on SimSat have a rotor rate  $\Omega$  of 270 rad/sec, while  $\dot{\delta}_j$  is restricted to 2.5 rad/sec, validating Eq. (2.40) which in turn implies that Eq. (2.41) will be true for except when  $\dot{\delta}_j$  is zero. Applying the assumptions in Eqns. (2.40) and (2.41) reduces Eq. (2.37) to

$$\begin{aligned} \dot{\vec{H}}_{net} = \mathbf{I}_b \dot{\vec{\omega}}_{bi} + \sum_{j=1}^4 \left( \mathbf{R}_3(\theta_j)^T \begin{bmatrix} -I_R \Omega \sin(\beta) \sin(\delta_j) \dot{\delta}_j \\ -I_R \Omega \cos(\delta_j) \dot{\delta}_j \\ -I_R \Omega \cos(\beta) \sin(\delta_j) \dot{\delta}_j \end{bmatrix} \right) + \\ \vec{\omega}_{bi} \times \mathbf{I}_b \vec{\omega}_{bi} + \vec{\omega}_{bi} \times \sum_{j=1}^4 \left( \mathbf{R}_3(\theta_j)^T \begin{bmatrix} I_R \Omega \sin(\beta) \cos(\delta_j) \\ -I_R \Omega \sin(\delta_j) \\ I_R \Omega \cos(\beta) \cos(\delta_j) \end{bmatrix} \right). \end{aligned} \quad (2.40)$$

Equation (2.40) can be further simplified by factoring out like terms, removing the summations, and converting to matrix form, resulting in

$$\dot{\vec{H}}_{net} = \mathbf{I}_b \dot{\vec{\omega}}_{bi} + I_R \Omega \mathbf{A} \begin{bmatrix} \dot{\delta}_1 \\ \dot{\delta}_2 \\ \dot{\delta}_3 \\ \dot{\delta}_4 \end{bmatrix} + \vec{\omega}_{bi} \times \mathbf{I}_b \vec{\omega}_{bi} + \vec{\omega}_{bi} \times I_R \Omega \bar{h}_{cmga} \quad (2.41)$$

where

$$\mathbf{A} = \begin{bmatrix} -\sin(\beta) \cos(\delta_1) & \sin(\delta_2) & \sin(\beta) \cos(\delta_3) & -\sin(\delta_4) \\ \sin(\delta_1) & \sin(\beta) \cos(\delta_2) & -\sin(\delta_3) & -\sin(\beta) \cos(\delta_4) \\ \cos(\beta) \cos(\delta_1) & \cos(\beta) \cos(\delta_2) & \cos(\beta) \cos(\delta_3) & \cos(\beta) \cos(\delta_4) \end{bmatrix} \quad (2.42)$$

and

$$\bar{h}_{cmga} = \begin{bmatrix} -\sin(\beta) \sin(\delta_1) & -\cos(\delta_2) & +\sin(\beta) \sin(\delta_3) & +\cos(\delta_4) \\ -\cos(\delta_1) & +\sin(\beta) \sin(\delta_2) & +\cos(\delta_3) & -\sin(\beta) \sin(\delta_4) \\ \cos(\beta) \sin(\delta_1) & +\cos(\beta) \sin(\delta_2) & +\cos(\beta) \sin(\delta_3) & +\cos(\beta) \sin(\delta_4) \end{bmatrix}. \quad (2.43)$$

The vector  $\bar{h}_{cmga}$  is the combined directional component of the CMG array angular momentum, and  $I_R \Omega$  is the scalar magnitude of the angular momentum stored in a single, thus the combination of the two represents the total angular momentum of the CMG array

$$\vec{H}_{cmga} = I_R \Omega \begin{bmatrix} -\sin(\beta) \sin(\delta_1) & -\cos(\delta_2) & +\sin(\beta) \sin(\delta_3) & +\cos(\delta_4) \\ -\cos(\delta_1) & +\sin(\beta) \sin(\delta_2) & +\cos(\delta_3) & -\sin(\beta) \sin(\delta_4) \\ \cos(\beta) \sin(\delta_1) & +\cos(\beta) \sin(\delta_2) & +\cos(\beta) \sin(\delta_3) & +\cos(\beta) \sin(\delta_4) \end{bmatrix}. \quad (2.44)$$

The matrix  $\mathbf{A}$  represents the relationship between the change in the CMG array angular momentum with the gimbal rates. It can be shown through inspection of  $\mathbf{A}$  and  $\vec{H}_{cmga}$  that

$$\begin{aligned} I_R \omega \mathbf{A}(1, 1) &= \frac{\partial \delta(1)}{\partial H(1)} \\ I_R \omega \mathbf{A}(1, 2) &= \frac{\partial \delta(2)}{\partial H(1)} \\ &\vdots \\ I_R \omega \mathbf{A}(3, 3) &= \frac{\partial \delta(3)}{\partial H(3)} \\ I_R \omega \mathbf{A}(3, 4) &= \frac{\partial \delta(4)}{\partial H(3)} \end{aligned} \quad (2.45)$$

which is the definition for the Jacobian of  $\vec{H}_{cmga}$  with respect to  $\bar{\delta}$  or

$$\mathbf{J}_H = \mathbf{A} = \frac{\partial \bar{\delta}}{\partial \vec{H}_{cmga}} \quad (2.46)$$

and is a function of the gimbal angles. The Jacobian  $\mathbf{J}_H$  can then be used to relate the change in the CMG array angular momentum to the control variables by

$$\dot{\vec{H}}_{cmga} = \mathbf{J}_H \dot{\bar{\delta}} \quad (2.47)$$

where  $\dot{\vec{H}}_{cmga}$  is the controller solution and must be solved for  $\dot{\bar{\delta}}$ , however the Jacobian  $\mathbf{J}_H$  is not square and is therefore not directly invertible. Solutions to this equation, when applied to CMG arrays, are known as steering laws.

## 2.4 CMG Momentum Envelope and Steering Laws

Before addressing the steering laws, we will first consider the momentum envelope of the CMG array. The MATLAB program used to generate these momentum envelopes was based on algorithms sourced from Leve.[16] Before examining the full pyramidal CMG array, consider a single CMG. For a single CMG, the momentum envelope is a ring perpendicular to its gimbal axis, as shown in Fig. 2.18, where the black ring represents the momentum envelope and the line represents the gimbal axis.

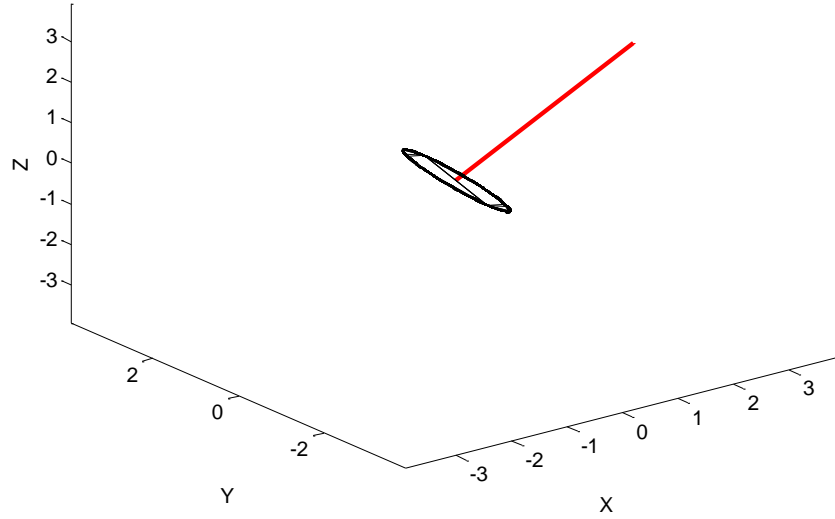
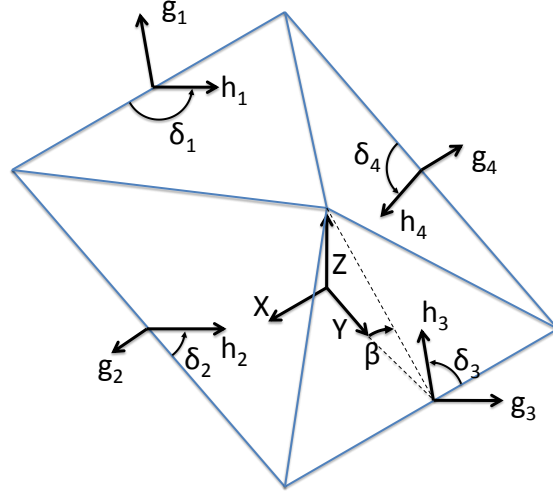


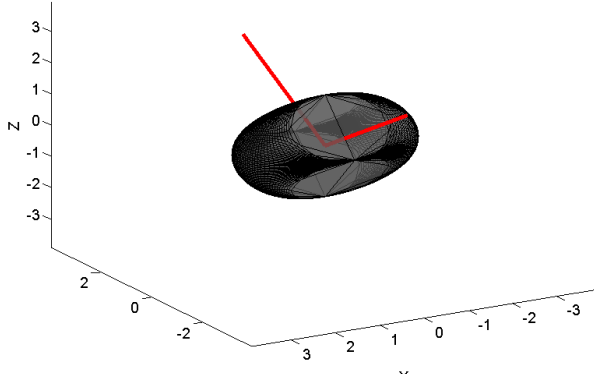
Figure 2.18: Angular Momentum Envelope for a Single CMG

Expanding to two CMGs, specifically those CMG #1 and #3, the envelope expands to a three dimensional surface, shown in Fig. 2.19. Figure 19(a) shows the alignment of the of the momentum planes. Figure 19(b) shows the envelope when the two momen-

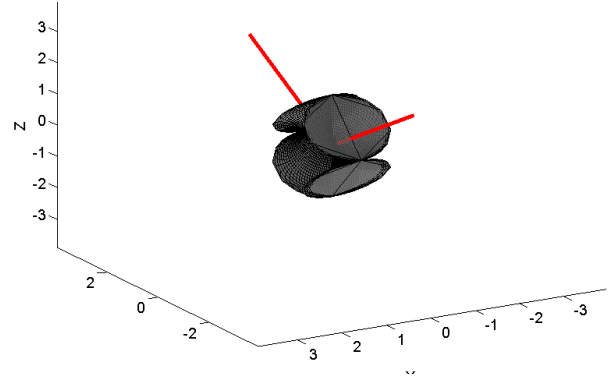
tum vectors are added together, forming the exterior singular surface, also known as the saturation surface. Figure 19(c) is the envelope when the momentum vectors are subtracted from one another and forms the interior singular surface. The combined angular momentum of the two CMGs is constrained to these two surfaces.



(a) 4-Unit Pyramid Configuration



(b) Exterior Momentum Envelope

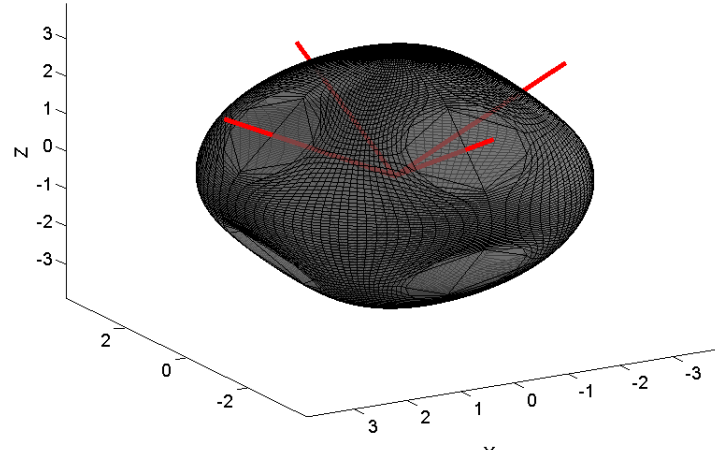


(c) Interior Momentum Envelope

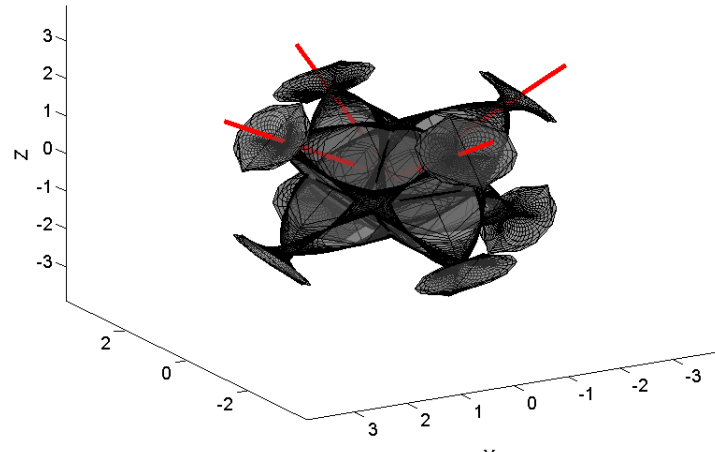
Figure 2.19: Angular Momentum Envelopes for a 2 CMGs Array

Expanding out to the four CMG pyramidal array, the possible angular momentum vector combinations change from a surface in three dimensional space into a volume. Instead, we will consider only the exterior and interior surfaces of this volume, where the CMG array reaches a maximum or minimum total angular momentum in that specified direction. The saturation surface, shown in Fig. 20(a) is found when each CMG's angular momentum vector is added together to provide the maximum angular momentum in the

specified direction. Momentum saturation is a physical constraint on the CMG array and must be addressed during the design process. Figure 20(b) shows the internal singularity surfaces, where one or more of the CMG's angular momentum is used to negate the others. Singularities along this internal surface cannot be pre-calculated and must be addressed in real-time during spacecraft operation.[16] The purpose of the steering law is to manipulate the angular momentum vector within this envelope to provide the desired torque, preferably without encountering a singularity.



(a) Exterior Momentum Envelope



(b) Interior Momentum Envelope

Figure 2.20: Angular Momentum Envelopes for a 4 CMG Array

Two steering laws were considered for implementation on SimSat, the Moore-Penrose Pseudoinverse Steering Law (MPPSL) and the Generalized Inverse Steering Law (GISL). These steering laws were considered for this research because their behavior is well understood and will serve to validate the CMG system performance without intro-

ducing the added complexity of an unproven steering law. Additionally, the null motion path was derived and used to verify the performance and behavior of the CMG array.

*2.4.1 Null Motion.* The Jacobian  $\mathbf{J}_H$  has more columns than rows, guaranteeing the existence of a null space, such that

$$\dot{\vec{H}}_{cmga} = \vec{0} = \mathbf{J}_H \dot{\vec{\delta}}_N \quad (2.48)$$

for  $\dot{\vec{\delta}}_N \neq 0$ . The null space results from having four actuators controlling a three dimensional problem. Applying  $\dot{\vec{\delta}}_N$  as the gimbal rates results in no change to the angular momentum CMG array and therefore imparts no torque on the vehicle. Ideally, the null space is one-dimensional, in actually this is not always true. At singularities, the null space exists mathematically, but null motion is physically unrealizable.[35]

*2.4.2 Moore-Penrose Pseudoinverse Steering Law.* The first steering law implemented in this research was the Moore-Penrose Pseudoinverse Steering Law (MPPSL), also known as the right-inverse, is a method of solving the system

$$\mathbf{A}\bar{x} = \bar{b} \quad (2.49)$$

where  $\mathbf{A}$  has more columns than rows by creating the Moore-Penrose Pseudoinverse matrix

$$\mathbf{A}^+ = \mathbf{A}^T (\mathbf{A}\mathbf{A}^T)^{-1} \quad (2.50)$$

and using  $\mathbf{A}^+$  to solve

$$\bar{x} = \mathbf{A}^T (\mathbf{A}\mathbf{A}^T)^{-1} \bar{b} = \mathbf{A}^+ \bar{b}. \quad (2.51)$$

$\mathbf{A}^+$  exists if and only if  $\mathbf{A}$  has a rank equal to the number of rows.[35] Applying the Moore-Penrose Pseudoinverse to Eq. (2.47) yields

$$\dot{\vec{\delta}} = \mathbf{J}_H^T (\mathbf{J}_H \mathbf{J}_H^T)^{-1} \dot{\vec{H}}_{cmga} = \mathbf{J}_H^+ \dot{\vec{H}}_{cmga} \quad (2.52)$$



where  $\dot{\vec{H}}_{cmga}$  is specified by the controller,  $\mathbf{J}_H$  is the Jacobian of the of the current CMG array, and  $\dot{\vec{\delta}}$  are the gimbal rates that will produce  $\dot{\vec{H}}_{cmga}$ . The MPPSL method does not address any of the internal singularities, and will fail any time  $\mathbf{J}_H \mathbf{J}_H^T$  is singular.

*2.4.3 Generalized Inverse Steering Law.* The Generalized Inverse Steering Law (GISL) is an alternative method of inverting  $\mathbf{J}_H$  that has fewer singularities, specifically it avoids non-degenerate singularities by coupling in null motion with the gimbal movements. It cannot, however, avoid degenerate internal singularities, where null motion provides no escape path. The first step in generating the GISL is to define a new coordinate set in  $\bar{\Upsilon} \in \mathbb{R}^3$  such that

$$\dot{\vec{\delta}} = \mathbf{A}^T \dot{\bar{\Upsilon}}. \quad (2.53)$$

Therefore, substituting Eq. (2.53) into Eq. (2.47) yields

$$\begin{aligned} \dot{\vec{H}}_{cmga} &= \mathbf{J}_H \dot{\vec{\delta}} \\ &= \mathbf{J}_H \mathbf{A}^T \dot{\bar{\Upsilon}}. \end{aligned} \quad (2.54)$$

If we assume  $\mathbf{J}_H \mathbf{A}^T$  is non-singular, then

$$\dot{\bar{\Upsilon}} = (\mathbf{J}_H \mathbf{A}^T)^{-1} \dot{\vec{H}}_{cmga} \quad (2.55)$$

which can be substituted back into Eq. (2.53) which provides the GISL as

$$\dot{\vec{\delta}} = \mathbf{A}^T (\mathbf{J}_H \mathbf{A}^T)^{-1} \dot{\vec{H}}_{cmga}. \quad (2.56)$$

If  $\mathbf{J}_H$  is used for  $\mathbf{A}$ , then the GISL simplifies to the Moore-Penrose solution that was previously derived. If, however, a different  $\mathbf{A}$  matrix is used then the solution can be more robust. Asghar and Leve both propose that

$$\mathbf{A} = \mathbf{J}_H + \mathbf{D} \quad (2.57)$$

where matrix  $\mathbf{D}$  has column vectors that are orthogonal to the columns in  $\mathbf{J}_H$ . One option is to define  $\mathbf{D}$  as

$$\mathbf{D} = I_R \Omega \begin{bmatrix} -\sin(\beta) \sin(\delta_1) & -\cos(\delta_2) & \sin(\beta) \sin(\delta_3) & \cos(\delta_4) \\ -\cos(\delta_1) & \sin(\beta) \sin(\delta_2) & \cos(\delta_3) & -\sin(\beta) \sin(\delta_4) \\ \cos(\beta) \sin(\delta_1) & \cos(\beta) \sin(\delta_2) & \cos(\beta) \sin(\delta_3) & \cos(\beta) \sin(\delta_4) \end{bmatrix} \quad (2.58)$$

which can be substituted into Eq. (2.56), providing the complete GISL as

$$\dot{\delta} = (\mathbf{J}_H + \mathbf{D})^T \left( \mathbf{J}_H (\mathbf{J}_H + \mathbf{D})^T \right)^{-1} \dot{\vec{H}}_{cmga}. \quad (2.59)$$

TD was chosen specifically because it forces null motion behavior into the specified control solution. The GISL, defined in this manner, can avoid/pass through non-degenerate internal singularities by coupling in null motion, but still suffers from degenerate internal singularities and saturation. This will occur any time  $\mathbf{J}_H (\mathbf{J}_H + \mathbf{D})^T$  is singular.[28, 16]

## 2.5 Linearized Proportional-Integral-Derivative Attitude Control

*2.5.1 PID Control.* The proportional-integral-derivative (PID) controller is a linear control algorithm that is widely used because of its of simple implementation and applicability to a wide range of control problems. The PID controller is particularly useful if the plant dynamics are either poorly understood or poorly characterized. When tuned properly, PID control can provide satisfactory, but not necessarily optimal, control for a range of inputs and disturbances. Because of these properties, PID control, as well as proportional-integral and proportional-derivative controls are used in over half of the industrial controllers used today.[24]

PID control operates on the difference between a desired state  $q_D$  and the actual state  $q_A$ , defined as the error  $q_E$ , to calculate a corrective action  $M_C$  which is fed to the plant  $G_P$ . Figure 2.21 shows a standard PID control block diagram. The components of the PID control are the proportional, integral, and derivative controllers. The proportional component produces a correction proportional to the, as dictated by the gain  $K_P$ . Increasing  $K_P$  yields in a larger response for a given error, resulting in a faster response,

but with more overshoot and a longer settling time. Additionally, very large values of  $K_P$  will result in instability, with ‘very large’ being application specific.

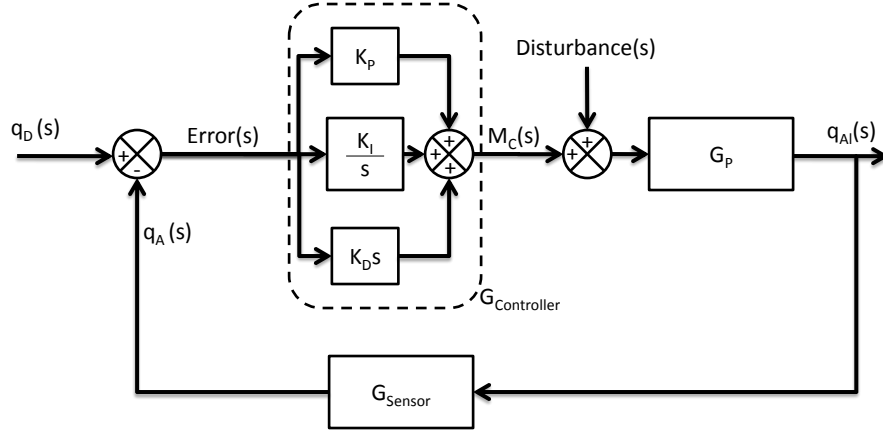


Figure 2.21: PID Controller

The integral component of the PID produces a response that is proportional to the magnitude and duration of the error, dictated by the gain  $K_I$ . The integral component, when combined with the proportional component, increases the response speed and more importantly eliminates the steady state error. The downside to the integral control is the increase in both overshoot and settling time due to integrator windup. Integrator windup occurs whenever there is a large change in error over a short timespan, such as a change in the desired state. Such a change requires a finite amount of time to correct, during which the integrator is accumulating the error, or winding up. The resulting control signal forces the system past the desired state while the integrator accumulates error in the opposite direction, or unwinds. Integral windup can be particularly troublesome in systems with limited control, because large changes in the desired state result in control saturation which further delays the response of the system.

The final component of the PID controller is the derivative component. The derivative component adds damping to the system as specified by the gain  $K_D$ . Increasing  $K_D$  decreases the overshoot and settling time of the system, while also slowing the response. Derivative control is most commonly used to reduce the overshoot caused by large proportional and integral gains. Unfortunately, numerical differentiation greatly amplifies

any noise in the error signal, which is further amplified by large  $K_D$ , and can cause instability in a system that is theoretically stable. Therefore, derivative control is most often implemented using either an approximate derivative or a sensor that can measure error derivatives directly rather than differentiating the error signal. As an example, for attitude control the rate gyroscope directly measures body rates, which are directly related to the derivative of the orientation, as shown in Section 2.3.1.

*2.5.2 Linearized Attitude Control.* The PID algorithm is a linear controller and is not assured to stabilize a system with the nonlinear kinematics inherent in quaternions nor the nonlinear dynamics Euler's equation, Eq. (2.17). As stated in Section 2.3.1, quaternions are approximately linear on a range of  $\pm 0.6$  radians. Because the controller operates on the difference between the desired and actual orientation, the error will likely be within this bound and thus can be assumed to be linear. The nonlinear dynamics, however, must be addressed, which can be done via feedback linearization. Feedback linearization involves calculating the nonlinear terms in the dynamics and incorporating them into the control signal to cancel out the nonlinearity, as shown in Fig. 2.22. Using

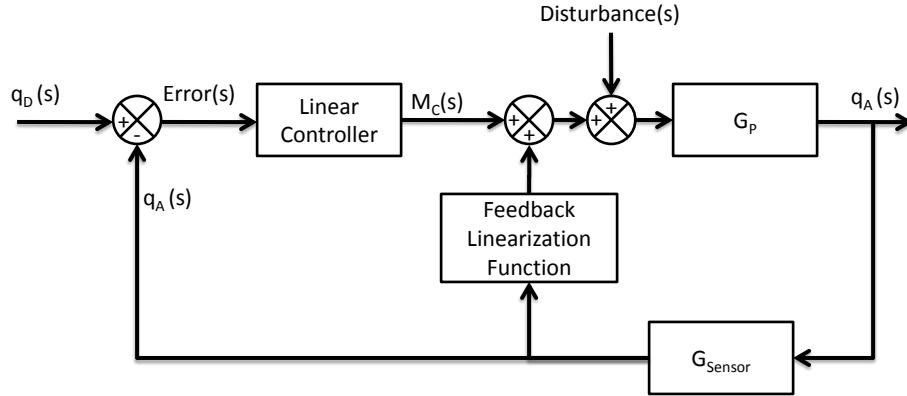


Figure 2.22: Feedback Linearization

Euler's equation, restated here as Eq. (2.60)

$$\vec{M} = \mathbf{I}\dot{\vec{\omega}}_{bi} + \vec{\omega}_{bi} \times \mathbf{I}\vec{\omega}_{bi} \quad (2.60)$$

then the applied moment  $\vec{M}$  is the control variable,  $\vec{\omega}_{bi} \times \mathbf{I}\vec{\omega}_{bi}$  is the nonlinear term that must be addressed, with the goal of having  $\dot{\vec{\omega}}_{bi}$  linearly related to control input. In order to address  $\vec{\omega}_{bi} \times \mathbf{I}\vec{\omega}_{bi}$ , the control term  $\vec{M}$  is split into two components, as shown in Fig. 2.22, or

$$\vec{M} = \vec{M}_{LC} + \vec{M}_{NLC} \quad (2.61)$$

where  $\vec{M}_{LC}$  is the control signal from the linear controller and  $\vec{M}_{NLC}$  is specified to cancel out the  $\vec{\omega}_{bi} \times \mathbf{I}\vec{\omega}_{bi}$ . Substituting Eq. (2.61) into Eq. (2.60) and setting  $\vec{M}_{NLC} = \vec{\omega}_{bi} \times \mathbf{I}\vec{\omega}_{bi}$  yields

$$\vec{M}_{LC} + \vec{\omega}_{bi} \times \mathbf{I}\vec{\omega}_{bi} = \mathbf{I}\dot{\vec{\omega}}_{bi} + \vec{\omega}_{bi} \times \mathbf{I}\vec{\omega}_{bi} \quad (2.62)$$

which simplifies to

$$\vec{M}_{LC} = \mathbf{I}\dot{\vec{\omega}}_{bi} \quad (2.63)$$

providing the desired linear relationship between  $\vec{M}_{LC}$  and  $\dot{\vec{\omega}}_{bi}$  which allows the use of a linear controller. Feedback linearization can also be used to address the nonlinear terms that result from having reaction wheels and CMG. Additionally, feedback linearization decouples the equations of motion, allowing the error in each axis to be addressed individually with its own PID controller, as shown in Fig. 2.23.[13]

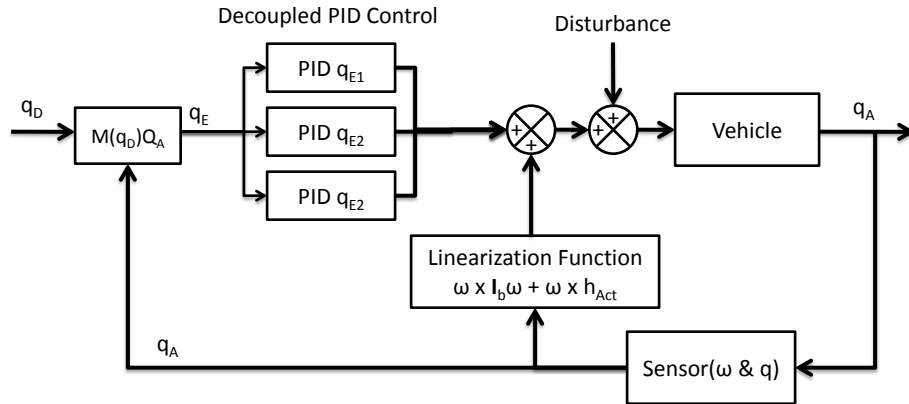


Figure 2.23: Three-Axis Linearized PID Controller

## **2.6   *Summary***

Chapter II covered the background information on satellite attitude dynamics simulators and AFIT's past work in the field. Next, spacecraft dynamics were explored with a focus on angular momentum exchange and CMG dynamics in support of the development efforts that are presented in Chapter III. Finally, linearized feedback control theory was using PID controllers was presented.

## III. Methodology

### 3.1 Introduction

Chapter III covers the design, construction, integration, software development, and validation of the CMG array for SimSat. The first two sections introduce the baseline SimSat hardware and software used in the fan/thruster and reaction wheel ACS which the CMG array integrates with. The third section covers changes and improvements to the reaction wheel actuators. The next section covers the design of the CMG array through hardware integration and calibration. The fifth section covers CMG software development and integration. The final section of Chapter III presents the tests performed to validate the performance of the hardware.

### 3.2 SimSat System Baseline Hardware

In order to design a CMG array to interface with SimSat, it was necessary to understand the baseline hardware configuration. The CMG array had to physically fit on SimSat's deck and interface with the existing on-board control system. Section 3.2 will introduce the subsystems relevant to the CMG array design. The SimSat spacecraft dynamics and control testbed designed and built by Roach *et al.* consists of three major hardware systems [25]:

1. A ground station (Figure 3.1)
2. A tri-axial air-bearing (Figure 3.2)
3. SimSat vehicle (Figure 1.1)

*3.2.1 Ground Station and Data Link.* The ground station for SimSat consists of a custom built Windows computer, using a commercial 802.11b wireless data link to communicate with SimSat. Figure 3.1 shows the ground station in its current configuration. The ground station's primary purpose is to provide remote access to SimSat's on-board computers to command the vehicle, monitor its operations, and record data. It does this through either Windows remote desktop to access the on-board computers, or through custom MATLAB communications programs. The ground station is also used for both real-time and post processing of the data collected during experiments.

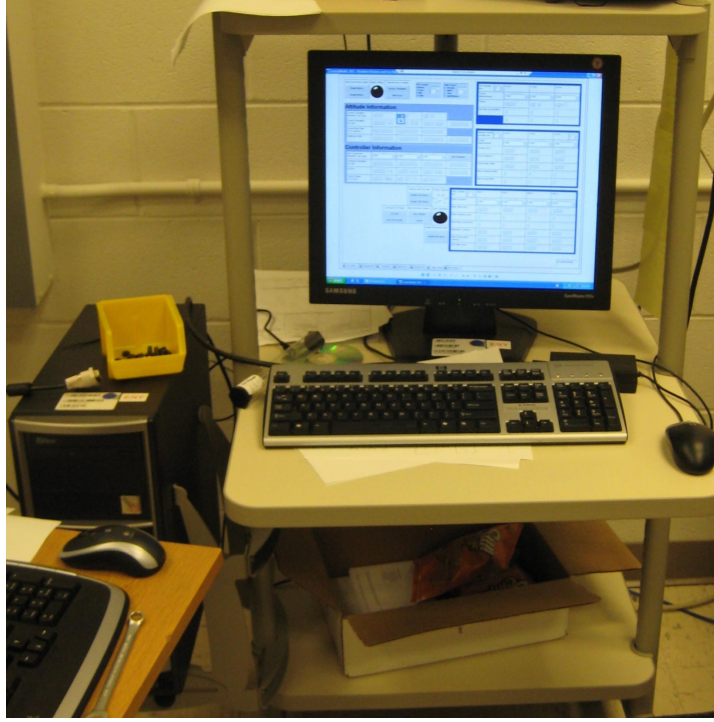
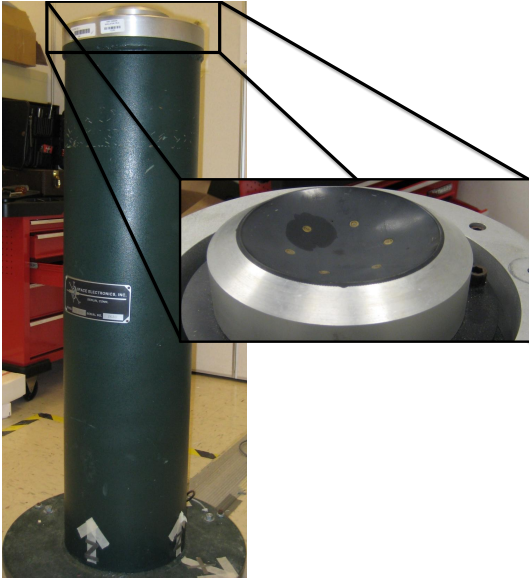


Figure 3.1: SimSat Groundstation

*3.2.2 Air-Bearing.* SimSat uses the Space Electronics, Inc. model SE-9791 tri-axial spherical air-bearing to support the vehicle and provide a near torque free environment. This model of air-bearing consists of a precision rotor and stator with equal radii of curvature and separated by a cushion of air which is less than  $13\text{ }\mu\text{m}$  thick. The cushion of air is maintained through jewel orifices that meter the air flow and provide dynamic centering of the rotor [1]. The system specifications and stator, housing the jewel orifices atop the pedestal, are shown in Fig. 3.2.

SimSat is configured as tabletop satellite simulator, with the air-bearing at the geometric center of the main deck. This design allows SimSat to have unconstrained motion about its  $z$ -axis. Motion about the  $X$ - and  $Y$ -axes are restricted to  $\pm 30^\circ$ , with a support ring to limit movement beyond this range. The vehicle is balanced prior to each test to keep its center of mass coincident with the center of rotation. Balancing techniques are described in detail in Section 3.7.1 later in this chapter.





Ball Bearing Diameter	22.00	cm
Pedestal Cup Diameter	5.72	cm
Unloaded Ball Bearing Mass	19.05	kg
Maximum Loading	136.08	kg

Figure 3.2: Space Electronics, Inc. Tri-Axial Spherical Air-Bearing

*3.2.3 SimSat Vehicle Hardware.* The baseline SimSat vehicle contains four major subsystems which were of significant concern for the design and development of the CMG array. These systems are

1. Mini-Box PC (Figure 3.3)
2. dSPACE MicroAutoBox (Figure 3.4)
3. Northrop Grumman LN-200 Fiber Optic Gyroscope IMU (Figure 3.5)
4. Reaction Wheel and Fan/Thruster Actuators (Figure 3.6)

For additional information on the subsystems listed in this subsection, as well as any subsystems not discussed, refer to the thesis work of Roach *et al.* and Snider.[25, 34]



Figure 3.3: Mini-Box PC

*3.2.3.1 Mini-Box PC<sup>®</sup>.* The Mini-Box PC serves three functions. The first is to provide the SIMULINK software development interface used to program the dSPACE MicroAutoBox. In this role, the Mini-Box PC converts user supplied SIMULINK models to compiled real-time programs, and loads the compiled programs onto the MicroAutoBox for execution. The second function the Mini-Box PC performs is to interface with the MicroAutoBox during real-time operations using the dSPACE ControlDesk or the dSPACE mLib MATLAB interface. ControlDesk and mLib are discussed in more detail in Sections 3.3.4 and 3.3.5, respectively. The final function of the Mini-Box PC is to host SimSat's 802.11b interface to provide a telemetry link to the ground station. This can be performed using Windows remote desktop, or through custom MATLAB communications programs.

*3.2.3.2 dSPACE MicroAutobox.* The dSPACE MicroAutoBox is a programmable, real-time processor and data acquisition computer that executes programs supplied by the attached Mini-Box PC. The MicroAutoBox serves as the command and data handling processor on SimSat. The MicroAutoBox also provides data interfaces for the various hardware on SimSat. These interfaces include the Controller Area Network (CAN) bus, RS-232 port, analog, and digital I/O ports which MicroAutoBox uses to control SimSat's attitude. The physical characteristics of the MicroAutoBox are shown in Fig. 3.4.



Weight	2.15	kg
Width	182	mm
Length	192.6	mm
Height	50	mm
Power Consumption	30	W

Figure 3.4: dSPACE MicroAutoBox



Figure 3.5: Northrop Grumman LN-200 Fiber Optic Gyroscope IMU

*3.2.3.3 LN-200 Fiber Optic Gyroscope IMU.* The Northrop Grumman LN-200 IMU is a strap-down inertial measurement platform that provides the MicroAutoBox with attitude information. The LN-200 contains three orthogonally mounted fiber-optic gyroscopes. Each gyroscope has two coils of fiber-optic cable wound in opposite directions, a laser diode, and an interferometer. As the gyroscope rotates, one laser path length effectively becomes longer while the other becomes effectively shorter because the speed of light is independent of the movement of the fiber-optic cable. The change in length creates a phase shift known as the Sagnac effect which is measured by the interferometer to provide angular rates.[29] The LN-200 also contains three orthogonally mounted accelerometers which can be used to detect accelerations acting on the IMU, notably the acceleration due to gravity. The accelerometers are also sensitive to vehicle vibrations which was a concern when designing the CMG array mounting system. A custom electronics box converts the LN-200's proprietary signals at 400 Hz to RS-485 serial protocol. The electronics box also contains a SkEye Unlimited RS-485 to RS-232 interface board that down-samples the signal to 200 Hz and provides the measurements to the MicroAutoBox.

*3.2.3.4 Reaction Wheel and Fan/Thruster Actuators.* The baseline SimSat had two sets of attitude control actuators, six orthogonal fan/thrusters and three orthogonal reaction wheels. Both systems use Maxon EC brush-less DC motors equipped with integrated shaft encoders. The motors are controlled by Maxon EPOS 70/10 motor control units, shown in Fig. 3.6. The EPOS unit supports multiple modes of operation; velocity mode is used for both the fans and reaction wheels. In velocity mode,

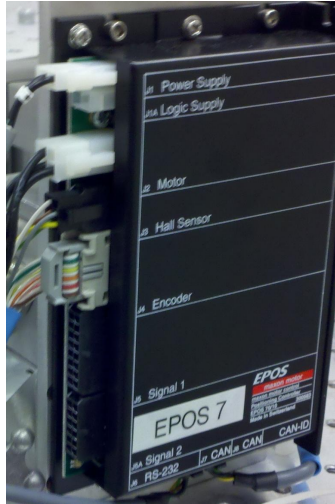


Figure 3.6: Maxon EPOS Motor Controller

the EPOS uses a closed loop PI regulator to achieve and maintain the specified motor speed. The EPOS units communicate with the MicroAutoBox using the CAN bus using the CANOpen communications protocol. More information on CANOpen is provided in Section 3.3.2. The CAN bus is easily expandable, with support for up to 127 devices. The EPOS controller supports the full range of Maxon EC motors, and is configurable through either the CAN bus or the proprietary software.

### 3.3 *SimSat Software Applications*

SimSat uses multiple software applications to support both hardware diagnostics and experiment development. Section 3.3 will address the key software applications used to for command and control on SimSat. For this research, the SimSat spacecraft dynamics and control testbed relied primarily upon five software applications:

1. EPOS User Interface<sup>®</sup>
2. CANOpen Communications Protocol
3. SIMULINK<sup>®</sup>
4. dSPACE ControlDesk<sup>®</sup>
5. MATLAB with dSPACE mLib

**3.3.1 EPOS User Interface®.** The EPOS user interface (UI) is resident on the Mini-Box PC. It is a proprietary software application provided by Maxon Motor for use with their EPOS line of controllers. The EPOS UI is used to configure the EPOS controller to the user's specifications.[25] The EPOS UI communicates with the EPOS units via the serial port on the Mini-Box PC and does not interact with the MicroAutoBox. The key configuration parameters include motor specifications, CANOpen communications parameters, and internal controller gains. These configuration parameters are stored in the CAN Object library. The EPOS UI also provides auto-tuning algorithms to assist in determining the optimal controller gains. The auto-tune algorithm performs a series of step-inputs and adjusts the gains to minimize overshoot, rise time, and settling time. Figure 3.7 shows the auto-tuning results run on a fan/thruster actuator. The auto-tuning algorithm was used to verify calibration of the fan/thrusters. The reaction wheels were tuned and verified using the manual techniques laid out in Section 3.4.2 of Snider's thesis.[34]. Lastly, the EPOS UI is also used as a diagnostic tool read error messages and manually activate each motor.

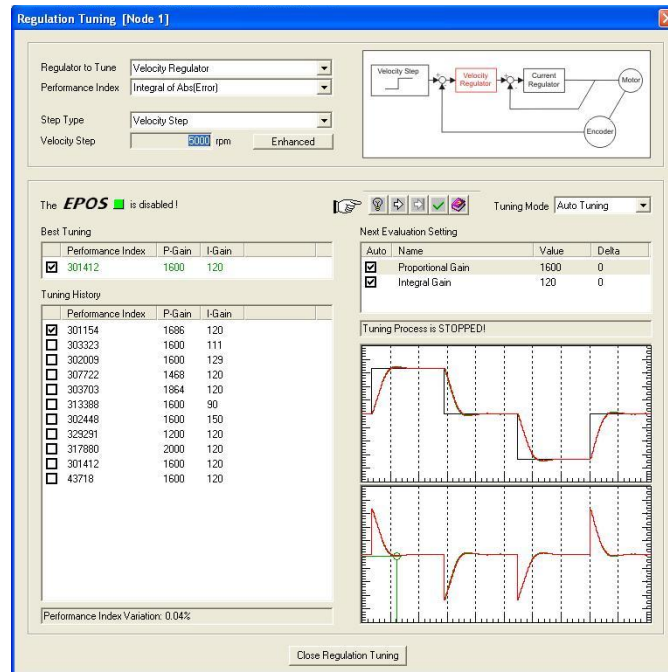


Figure 3.7: EPOS UI Velocity Auto-Tuning Program

*3.3.2 CANOpen Communications Protocol.* The CANOpen Communications Protocol is used on SimSat to connect the MicroAutoBox to the EPOS 70/10 motor controllers. CANopen is the internationally standardized (EN 50325-4) CAN-based higher-layer protocol for embedded control system. On SimSat, the MicroAutoBox serves as the CAN master node, and the EPOS units and G0AMH encoders serve as a CAN slave nodes. All slave nodes have a unique identifier. Each CAN slave maintains a database, known as the object library, which contains the configuration and operational parameters for the controller. The EPOS controllers store motor specifications such a maximum current, configuration parameters such as control gains, and operational parameters such as desired speed. The object library also stores sensor information gathered from the controller, such as measured speed. The controller updates and responds to the object library in real-time.

The CANOpen protocol allows the master controller to manipulate the object libraries of the slaves it shares a connection with. There are two methods for transferring information, the service data object (SDO) and the process data object (PDO). The SDO method allows the master full access to a slave's object library. SDO messages always come in pairs, the master sends a message, and the slave responds. SDO outbound message consists of the SDO message identifier, command byte, a three byte object library address, and up to four bytes of data. The command byte specifies read or write, along with how much data is being transferred. In a write operation, the data is copied to the specified object library address; in a read operation the data at the object library address is sent back in the response message. The SDO response consists of a message identifier, status byte, object library address, and up to four data. If an error occurs, the status byte indicates why the error occurred.

The SDO method is primarily used for configuration and setup. All commands have a confirmation or error message, allowing for system protection. Additionally, some areas of the object library, like resetting faults, can only be accessed using SDO. Figure 3.8, illustrates the SDO communications mode. The SDO method does have two major drawbacks. First, each node has a single address, limiting communication to four bytes per transfer cycle per node and no simultaneous commands. Second, because all traffic

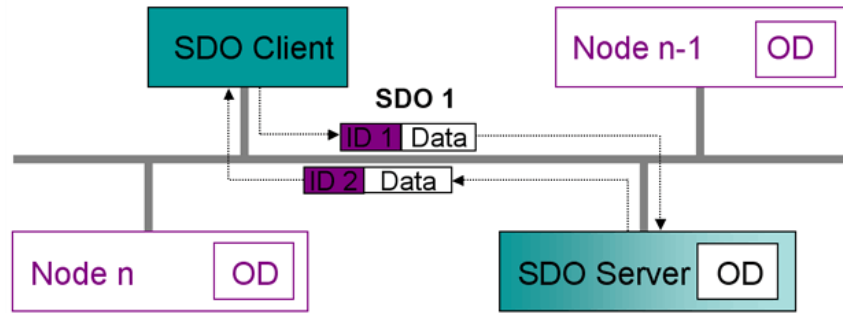


Figure 3.8: CANOpen SDO Protocol

is bidirectional, each message uses 20 bytes of bandwidth. In practice, this prevents simultaneous communication between the master and multiple slaves.

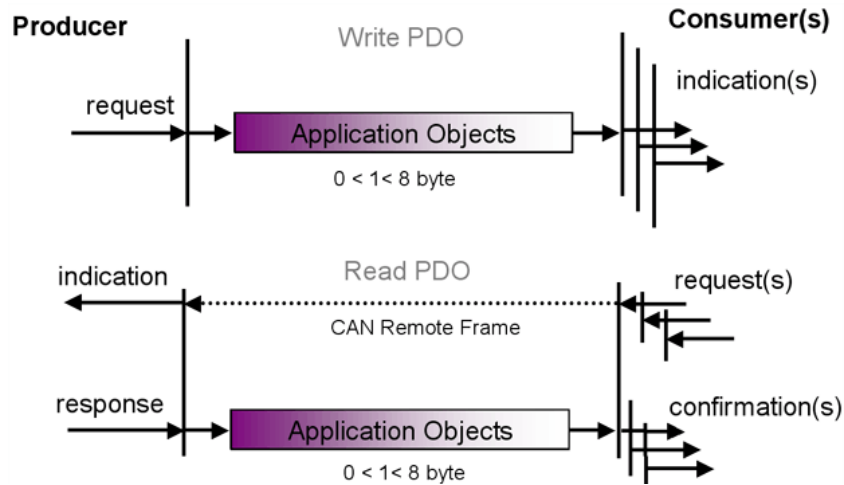


Figure 3.9: CANOpen PDO Protocol

The second data transfer method, the PDO, is used to monitor and update the control process. There are two types of PDOs; transmit and receive. Both types of PDOs are configured using SDO messages to specify what object library entries will be manipulated. Transmit PDOs send data from the slave's object library to the master. This can be triggered internally based on sensor conditions and the system timer, or externally from the system master via a remote transmission request (RTR). Receive PDOs accept data from the master and place it in the object library. Figure 3.9 illustrates the PDO communications mode. Each PDO can transfer 8 bytes of data per message, and each node supports multiple transmit and receive PDOs. Additionally, PDOs do

not send confirmation messages, further reducing overhead. Finally, multiple nodes can be configured to respond to the same receive PDO, allowing multiple controllers to be manipulated simultaneously. The EPOS UI provides a software wizard to configure PDOs on the EPOS units as an alternative to using SDO messages for initial configuration. Once set, the PDO configuration is stored on-board the CANOpen device and reloaded at boot up.

**3.3.3 SIMULINK®.** SIMULINK is a commercially available software package from Mathworks designed for numerical simulation of dynamic systems. SIMULINK version 6.2 is installed on the Mini-Box PC, along with the SIMULINK Real-Time Workshop® and dSPACE MicroAutoBox programming and interface libraries. SIMULINK uses a graphical interface to streamline programming by building and connecting programming blocks, such as math functions, switches, and integrators. Figure 3.10 shows an example of a SIMULINK program written for SimSat. The dSPACE libraries provide software interfaces for the various input and output (I/O) ports on the MicroAutoBox. Programs are built in SIMULINK then the Real-Time Workshop to convert the model into C-code, which is compiled and loaded onto the MicroAutoBox via the dSPACE interface libraries. The compiled real-time program generated from the SIMULINK model runs in a fixed time-step mode with a 1 ms clock. This means all models built in SIMULINK execute a complete program loop every 1 ms, then return to the program start, and 1 ms is the smallest unit of time available in the program.

The SIMULINK model executed by the MicroAutoBox is responsible for several main functions. These are:

1. Collect data from on-board sensors
2. Maintain and update state information
3. Compute control solution
4. Convert control solution to actuator commands
5. Transmit actuator commands to the actuators





SIMULINK allows programming blocks to be grouped as reusable modules. Using modular code allows for rapid reconfiguration of SimSat’s control algorithms by changing the control blocks, but maintaining measurement and actuator blocks. Details of the SIMULINK program developed during this research are provided in Section 3.6

*3.3.4 dSPACE ControlDesk®.* The dSPACE ControlDesk program is used to access and manipulate data within the real-time program being executed by the MicroAutoBox. ControlDesk provides developers with a workspace to build experiment specific graphical user interfaces which are linked to the SIMULINK model and its corresponding program on the MicroAutoBox.[5, 6] Figure 3.11 shows an example layout used to test the CMG array during its development. ControlDesk is the primary interface software to SimSat when using Windows remote desktop from the ground station and was used to initialize the system and conduct initial experiment runs.

*3.3.5 MATLAB® with dSPACE mLib.* In addition to ControlDesk, dSPACE provides its mLib library for use in MATLAB. The mLib library allows MATLAB programs and scripts on the Mini-Box PC to access and manipulate data within the real-time program on the MicroAutoBox. Additionally, mLib allows for windowed data capture, where a fixed number of samples are recorded on the MicroAutoBox which are then downloaded to the Mini-Box upon completion, allowing for simultaneous capture of multiple variables.[5, 6] The primary use of mLib was to automate experimental procedures to provide repeatability, reduce execution time, and minimize human error. Additionally, once data is in the MATLAB environment on the Mini-Box PC it is available for immediate post-processing and analysis or can be transferred to other computers in real-time using communications functions supplied with MATLAB.

### **3.4 Reaction Wheel Actuator Changes**

Several changes were made to the reaction wheel actuators to address ‘sporadic control’ issue and improve the momentum storage capabilities of the reaction wheel subsystem.[34]

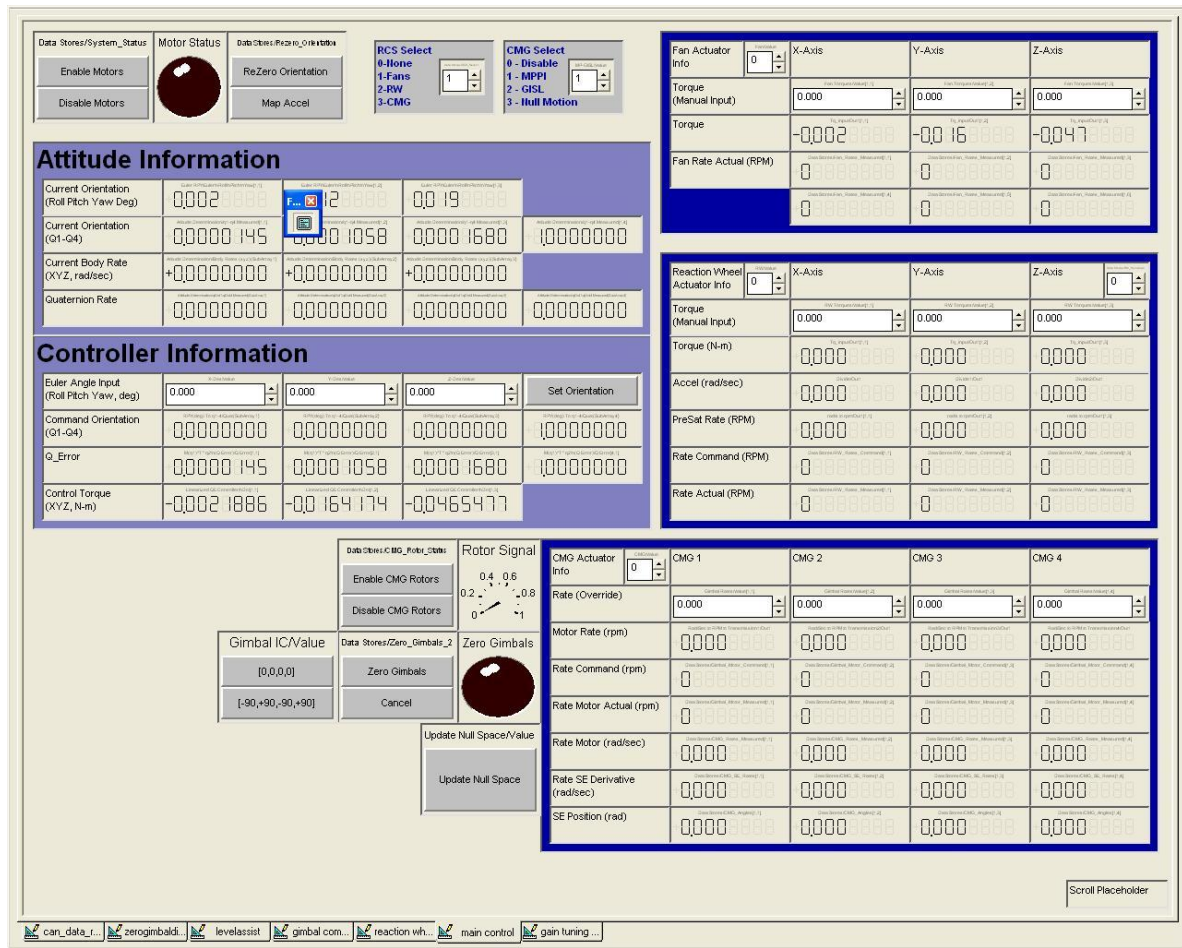


Figure 3.11: Example dSPACE ControlDesk Layout

**3.4.1 20cm Reaction Wheel.** The original reaction wheel design was a 10.16 cm diameter stainless steel (hereafter referred to as the 10 cm wheels) with an MOI of 0.00261 kg-m<sup>2</sup> and 0.004 kg-m<sup>2</sup> for the 2.54 cm and 5.08 cm thick wheels, respectively. In order to have sufficient MOI using 10 cm stock, the wheels are a solid design, and therefore have a poor MOI to mass ratio of  $1.719 \times 10^{-3} \text{ m}^2$ . The 10 cm wheels were sufficient to test and validate the reaction wheel motors and algorithms, but limited the overall capability of the system. New wheels were designed that have 20.32 cm diameter using spokes to maximize the MOI to mass while increasing the MOI by approximately four times vs. the 10 cm wheels. Both the 10 cm and 20 cm are shown in Fig 3.12.

The new 20 cm reaction wheels were designed to maximize rotational inertia about the axis of rotation while minimizing the mass. The ideal design is a large, thin ring; however, in practice this design is difficult to manufacture. The design implemented is



Figure 3.12: 10 cm and 20 cm Reaction Wheels

composed of a hub, 6 spokes, and a metal ring. The new wheels were sized to provide approximately double the MOI of the 5.08 cm thick wheel, or four times the MOI of the 2.54 cm thick wheels by adjusting the thickness of the metal ring. As a safety consideration, a sheet of plexiglass was inserted into the recessed area of the wheel to prevent objects from accidentally getting caught in the spokes. This plexiglass also minimized the airflow around the wheel spokes.

Finally, the wheels were precision balanced for operations up to 9000 rpm. This process removed small amounts of material at key points to minimize lateral vibrations. While small, these imbalances would, quite literally, shake the actuator or Simsat to pieces and posed a significant safety risk.

The MOI of each wheel was measured using AFIT's MOI measurement system, a Space Electronics XR250. The XR250 uses a torsion spring oscillator to measure the MOI about the axis of rotation. The instrument is accurate to  $0.25\% + 2.9 \text{ g-cm}^2$ . Mass was measured using the a Scout Pro digital scale, accurate to 0.1 g. Table 3.1 lists the

measurements for each reaction wheel. The MOI of the EC-45 reaction wheel motor's axle shaft is 119 g-cm<sup>2</sup>.

Table 3.1: Reaction Wheel Characteristics

Wheel #	Mass (kg)	MOI (kg-m <sup>2</sup> )	MOI/Mass (m <sup>2</sup> )	Axis
1	1.1808	0.0076948	6.512x10 <sup>-3</sup>	Spare
2	1.2219	0.0079204	6.482x10 <sup>-3</sup>	Z-axis
3	1.2280	0.0079304	6.458x10 <sup>-3</sup>	Y-axis
4	1.2173	0.0079119	6.499x10 <sup>-3</sup>	X-axis

Initial testing of the 20 cm reaction wheels showed that despite the plexiglass, the spokes created a significant amount of air drag. While this is not a problem for the reaction wheel motor, it does create unstable, high speed air flow around the vehicle. This air flow places several unknown disturbance torques on the vehicle. To minimize these disturbances, acetate was glued to the front surface of the reaction wheel to create a smooth face and minimize turbulence. The acetate, combined with the plexiglass, prevented the spokes from creating excess turbulence.

*3.4.2 Electrical Subsystem Protection.* Electric motors are transducers which convert electrical energy into mechanical energy. The opposite is also possible, where the motor converts mechanical energy back into electrical energy. These devices are most commonly known as generators; however, there are many dual-use applications, such as regenerative braking technology and flywheel energy storage. The reaction wheel system on Simsat exhibited this behavior whenever a large rotor deceleration was commanded. The result was an uncontrolled voltage spike on the 37 V electrical bus and the destruction of an EPOS unit. This occurred using the original 10.16 cm diameter, 2.54 cm thick reaction wheels when commanding a stop from above 6000 rpm. The initial solution to the hard stop problem was to limit rotor speed changes in the software. In practice, rate limits had minimal impact on experiments. Large rotor accelerations and their associated torques generated were not used experimentally. The software solution did, however, prevent implementing 'emergency stop' commands. Additionally, any changes

to the reaction wheel software required updating and testing the safety protocols. A more permanent solution was needed.

The solution was to regulate the maximum electrical current draw to protect the computer system and add an electrical buffer to absorb or redirect the feedback energy away from the main electrical bus. Current regulation was provided by installing a  $2.7\ \Omega$  power resistor bank. When operating at nominal battery voltage of 37 V, the resistor bank limits continuous current draw to 13.7 A. Based on the battery's maximum output of 30 A, there is still 16 A available to run the computers and other actuators.

The Maxon Digital Shunt Regulator 70/30 (DSR) was chosen for the task of managing electrical feedback. The electrical block diagram for the DSR is shown in Fig. 3.13. The DSR has 8.8 mF of capacitance, allowing it to store 5.7 J of electrical energy. Should this storage be exceeded, the DSR shunts the energy to a block of resistors, where it is dissipated as heat. The DSR's capacitors also provide a limited ability to handle transient current loads in excess of the 13.7 A current limitations imposed by the resistor bank.

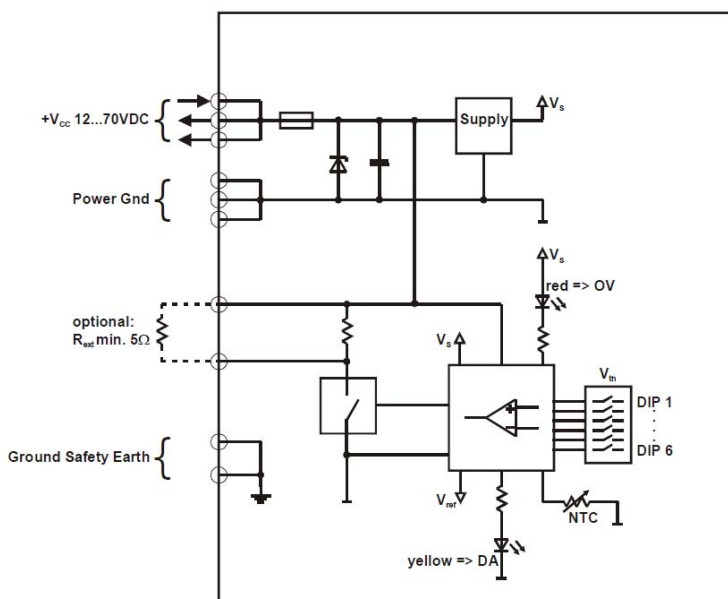


Figure 3.13: Maxon Digital Shunt Regulator 70/30 Block Diagram

### 3.5 *CMG Development and Sizing*

This section presents the design and development process for the SimSat’s CMG array from the system level requirements defined in Section 1.2 through assembly. The CMG design focused on meeting the desired system level requirements while minimizing time and cost to manufacture, to include consulting the machinists tasked to manufacture components.

*3.5.1 CMG Array Requirements.* The first step in sizing an attitude control actuator is to determine the approximate amount of torque the system will need to provide. The torque available determines disturbances the system can overcome and how quickly the system can react to commands. For a momentum exchange system, the total angular momentum storage available must also be specified, as this determines the saturation limits of the system. Once the system reaches saturation, it can no longer apply a torque in a particular direction and must rely on a different system for control. The maximum torque determines how hard or fast the system can accelerate while momentum storage determines how long the torque can be applied.

In order to estimate the torque requirements, the vehicle’s MOI needed to be estimated. Based on the work of Snider and Roach *et al.*, the Z-axis MOI and Z-axis maneuverability requirements drive the torque requirements on SimSat. The CMG system will add a significant, but undetermined amount of mass to the system. The increase in mass and MOI must be accounted for when estimating torque and momentum storage requirements. Based on the system growth experienced by Snider, as well as the planned reconfiguration of reaction wheels and counterbalance weights, the Z-axis MOI was estimated to increase from 10.1 kg-m<sup>2</sup> to around 15 kg-m<sup>2</sup>. The new MOI estimate is based on placing the CMG assemblies as close to the centerline as possible and using the reaction wheels as counterbalance to reduce the need for additional counterbalance mass. Using the estimated MOI, a peak torque of 0.5 N-m should provide an acceleration of 0.3 rad/sec<sup>2</sup>. Additionally, 0.5 N-m is more than sufficient to overcome any disturbance torques, which will be on the order of millinewton-meters.

CMGs are used primarily for their high torque to mass ratio and the associated reorientation capabilities that high torque can provide. As such, the momentum storage requirement is based on the maximum desired rotation rate of SimSat about the  $Z$ -axis. For SimSat  $4^\circ/\text{sec}$  or  $0.07 \text{ rad/sec}$  was chosen as an appropriate maximum  $Z$ -axis body rate. A  $0.07 \text{ rad/sec}$  allows SimSat to simulate the behavior of most CMG equipped vehicles which reach maximum design speeds of  $4\text{-}5^\circ/\text{sec}$ . A  $0.07 \text{ rad/sec}$  body rate requires approximately  $1 \text{ N-m-s}$  of angular momentum storage using SimSat's estimated  $Z$ -axis MOI of  $15 \text{ kg-m}^2$ . Alternatively, the momentum storage requirements could, and often are, sized based on the expected magnitude and duration of disturbance torque rejection.[4]

*3.5.2 CMG Number and Array Configuration.* The number and configuration of the CMG array determines the dynamic behavior of the CMG system, its associated control laws, and the sizing of the individual components. A single gimbal CMG with a fixed flywheel rate has only one degree of freedom, as such a minimum of three are required to provide 3-axis stabilization and control. In nearly all space applications, redundancy is desired, and thus the minimum number of CMGs increases to four, in a configuration that allows for any three to provide 3-axis control authority. Because SimSat should behave like an actual satellite and CMG dynamics are impacted by the configuration, the minimum number of CMGs on SimSat should be four. Increasing the number of CMGs can increase the redundancy or reduce the control complexity. Increasing the number of CMGs does; however, increase the mechanical complexity of the system, with associated impacts on mass, component sizes, and failure mechanisms. SimSat's physical size and available space limited the CMG array to a four unit design. As discussed in Section 2.3.5, this reduced the design space to the two primary configurations which use four CMGs are the pyramid and rooftop configuration.

The pyramid configuration, shown in Fig. 3.14, places each CMG along a horizontal plane equally spaced around a circle, with the momentum planes angle off the vehicle plane. The rooftop configuration places the CMGs in two parallel rows with the gimbal vectors forming a 'rooftop.' Both the pyramid and rooftop configurations allow for



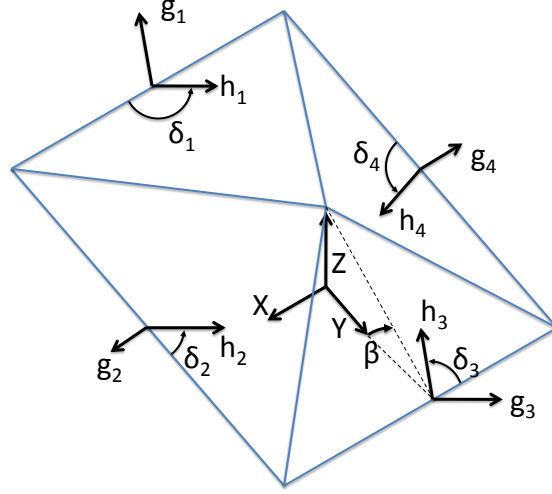


Figure 3.14: Four CMG Pyramid Configuration

tuning of the momentum and torque envelopes to meet the vehicle requirements by altering the angle of the momentum planes relative to one another. In both the pyramid and rooftop configurations, the  $\beta$  angle is measured relative to the vehicle's horizontal plane, indicated in Fig. 3.14. For SimSat, a uniform momentum and torque envelope was desired. With the pyramid configuration, a  $\beta$  angle of  $54.74^\circ$ , provides a near-spherical momentum envelope, as shown in Section 2.4. Additionally, the pyramid configuration has inherently better symmetry than the rooftop, making it a better fit for SimSat.

*3.5.3 CMG Rotor Sizing.* With the CMG array requirements and geometry locked down, the individual rotor requirements can be calculated. In order to obtain the



Figure 3.15: CMG Rotor

desired momentum envelope, each rotor assembly must store approximately 0.433 N-m-s. The rotor angular momentum requirements are based on the array torque requirements and the array geometry, using equations from Sec. 2.3.5. The amount of momentum stored in each rotor is dependent on its rotation rate and rotational inertia. A high speed electric motor is used to drive the rotor to speed and overcome bearing resistance. The Maxon EC-45 flat motor with integrated electronics was chosen for this application. The EC-45's integrated closed-loop controller helps ensure all four rotors run at the specified speed. The motors have an operating range of 0-6000 rpm, with 3000 rpm chosen as the design speed to ensure sufficient torque to maintain rotor speed with minimal variation. The rotor MOI requirements were calculated using Eq. (2.5) with an  $\omega$  of 3000 rpm to be 0.0014 kg-m<sup>2</sup>.

Stainless steel was chosen for the rotor material, based on its density of 8000 g/cm<sup>3</sup>, corrosion resistance, cost, and ease of machining. The maximum rotor diameter was limited to 10 cm based on the space available on SimSat and the availability of stock. Additionally, the length/height of the rotor was limited to about 5 cm to keep the gimbal assembly within a spherical envelope in order to minimize the impacts of asymmetry as described in Section 2.3.3. With the design constraints in mind, the main rotor was designed to maximize the rotational inertia, minimize the mass, and meet the required MOI. The design was also required to be stiff enough to not distort under the centripetal loads at 3000 rpm or any gimbal loads. Due to material and manufacturing defects, the rotors required high-speed, dynamic balancing to minimize vibration. Balancing removed small amounts of material to move the rotor's center of mass coincident with the center of rotation. Finally, the rotor was designed to interface with the motor and support bearings. The bearings chosen were size 608 high speed ball bearings. These bearings are single row, deep groove bearings which support both radial and axial loads. 608 bearings are commonly used on skateboards and are available in a wide variety of speed and load requirements. Figure 3.15 shows the final rotor configuration. The expected MOI of the rotor, prior to balancing, was 0.0017 kg-m<sup>2</sup>. After balancing, each rotor was measured using AFIT's XR250 for MOI measurements and Scout Pro scale for the mass measurements. These measurements are provided in Table 3.2. The difference from the

expected MOI was caused by both the removal of material and the 1.25% difference in density between the magnetic stainless steel (7900 g/cm<sup>3</sup>) used and the 316 stainless steel (8000 g/cm<sup>3</sup>) used in the initial calculations. Magnetic stainless steel was used based on its availability.

Table 3.2: CMG Rotor Characteristics

Wheel #	Mass (kg)	MOI (kg-m <sup>2</sup> )	% MOI Difference from Mean
1	1.0135	0.0016530	0.163607%
2	1.0089	0.0016431	-0.436284%
3	1.0168	0.0016622	0.721081%
4	1.0096	0.0016429	-0.448403%

*3.5.4 Gimbal Assembly Design.* The purpose of the gimbal assembly is to support the CMG rotor and allow it to be rotated relative to the rest of the vehicle. CAD was used in the design of the aluminum rotor housing to support the rotor, bearings, and motor. Figure 3.16 shows gimbal assembly with components labeled. One side of the rotor housing was designed to hold a Mercotac slip ring assembly to allow for unrestricted rotation of the gimbal. The other side of the housing attaches to the a steel shaft which connects the rotor housing to the gimbal motor. The gimbal motor bearings support the gimbal assembly on one side, and a bearing mounted on the slip ring assembly supports the other end.

In Section 2.3.3, the gimbal assembly was assumed to be symmetric about the gimbal axis; however, the gimbal assembly is clearly not symmetric. Because of this asymmetry, the MOI of SimSat will change as as the gimbal rotates. Additionally, if the center of mass of the gimbal assembly is not coincident with the gimbal axis then SimSat's center of mass will move as the gimbal rotates. In order to balance the assembly, a counterweight was installed opposite the rotor motor. Each gimbal assembly counterweight was matched to the gimbal assembly by placing the assembly on a level 'knife-edge' and reducing the counterweight mass until balance was achieved, as shown in Fig. 3.17. Once the gimbal assembly was balanced, the MOI was measured about the gimbal axis, rotor axis, and the third axis defined by the right hand rule using the AFIT's XR250. For each of these tests, the rotor was locked into place to prevent movement

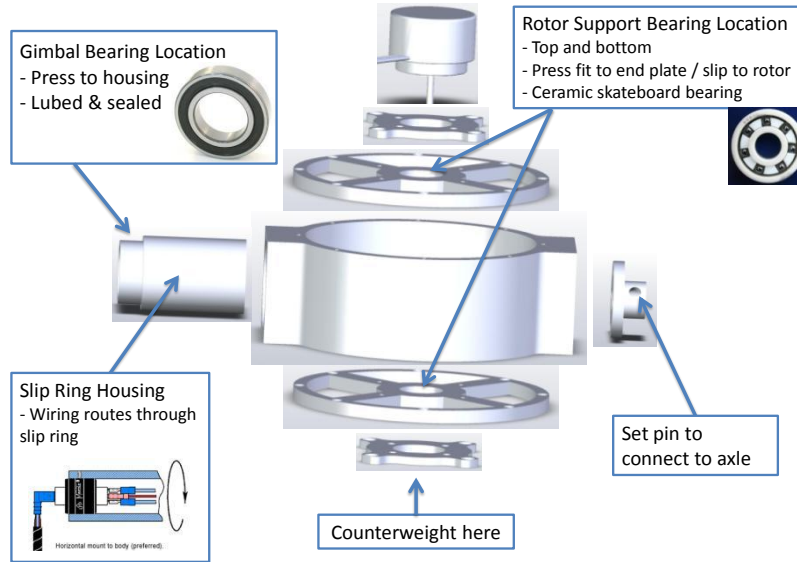


Figure 3.16: Gimbal Assembly

from distorting the measurements. The rotor axis MOI was measured at  $0.01177 \text{ Kg-m}^2$ , while the third axis was measured at  $0.01154 \text{ kg-m}^2$ , a difference of less than 2%. The difference small enough to neglect when compared to the overall MOI of SimSat which is on the order of  $10 \text{ kg-m}^2$  and the accuracy with which SimSat's MOI can be measured (see Section 4.3).

*3.5.5 Gimbal Motor Sizing.* The next step in designing the CMGs was to determine the appropriate motor to rotate the gimbal assembly. The motor was required to provide accurate performance at a low angular rate with minimal torque ripple. The maximum angular rate of the gimbal is expected to be around  $2 \text{ rad/sec}$  or  $20 \text{ rpm}$ . This low speed performance requirement was a significant issue, as most electric motors are not designed to operate effectively at low speeds, the exception being stepper motors. Stepper motors were discounted because they cannot deliver torque in a smooth fashion and have a limited angular resolution. Additionally, interoperability with the existing CANOpen architecture was desired. With these requirements in mind, Maxon Motors suggested using a brush-less DC motor with a reduction gearbox. The EC-MAX-30 series brush-less motor with a GP-32 159:1 reduction gearbox was selected, shown in Fig. 3.18. The motor is also equipped with an optical shaft encoder and is controlled using the

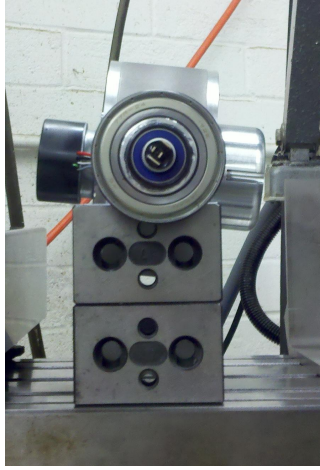


Figure 3.17: Gimbal Assembly Balancing

EPOS 70/10 controller. The optical shaft encoder mounted on the motor cannot be used to determine the gimbal angle because the GP-32 has approximately  $1^\circ$  of gear lash. The gear lash only affects position, not velocity, and therefore the EPOS velocity control mode should still be acceptable to control the gimbal velocity.[19, 20]

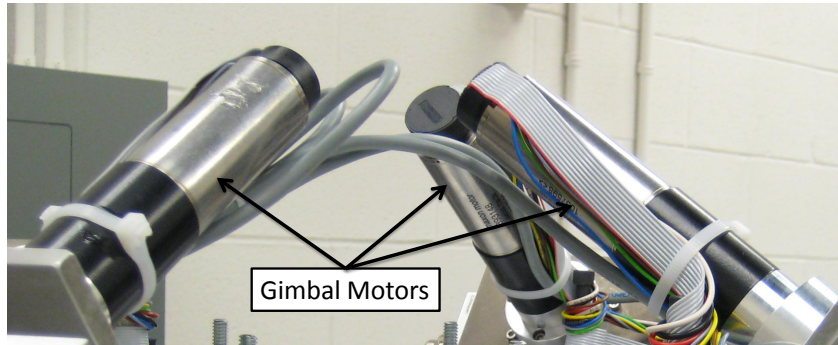


Figure 3.18: CMG Gimbal Motors

*3.5.6 Optical Shaft Encoder.* In order to accurately compute a control solution using a CMG array, the exact angle of each gimbal must be known. The EC-MAX-30 motor selected does provide position feedback, but the gear lash present in the GP-32 introduces  $1^\circ$  of uncertainty into the gimbal angle measurement which was deemed unacceptable; an alternate sensor was necessary. The gimbal angle sensor would need to directly measure the gimbal shaft to minimize errors due to gear lash in the gimbal motor transmission. The Baumer Electric G0AMH absolute encoder was selected for the task of accurately measuring the gimbal angle, shown in Fig. 3.19. The G0AMH uses

a 13 bit optical encoder, for a resolution of 8192 steps per turn or  $7.67 \times 10^{-4}$  rad/step. Additionally, the G0AMH has an integrated CANOpen interface, which allowed for easy integration with the MicroAutoBox. The encoder is mounted directly to the gimbal shaft, as shown in Fig. 3.19, to provide a direct gimbal angle measurement.

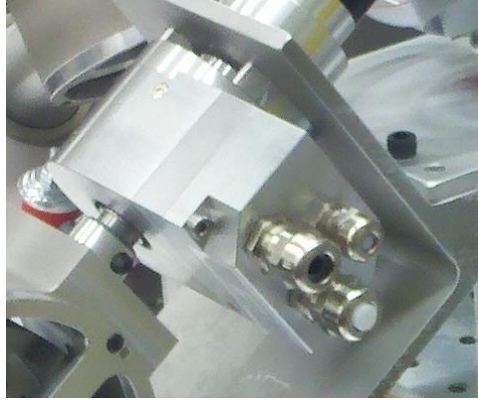


Figure 3.19: CMG Gimbal Encoder

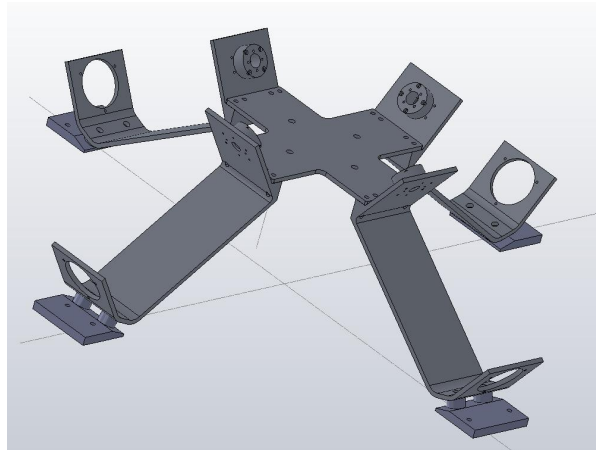


Figure 3.20: Gimbal Support Structure

*3.5.7 Gimbal Support and Vibration Isolation.* The gimbal support assembly holds the gimbal components in place on SimSat, sets the mounting angles  $\beta$  and  $\theta$ , and transfers loads generated by the CMGs to SimSat's structure. Each CMG gimbal held in a one-piece bracket and supported a roller bearing at one end and the gimbal motor's transmission. At the base of each bracket is a precision cut angle to match the angle of the mounting blocks to achieve a gimbal angle of  $35.26^\circ$ , resulting in the desired  $\beta$  angle of  $54.74^\circ$ . A similar angle was cut into the top of each bracket mount so to align each

bracket with a central, rigid cross brace. Figure 3.20 shows the CAD drawings of the gimbal support structure prior to construction.

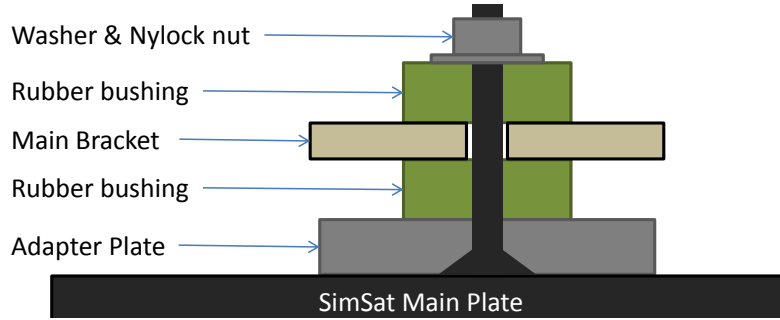


Figure 3.21: Vibration Isolation

The gimbal support assembly also contains vibration isolation. Because the CMG rotors are rotating at high speed, they will inherently produce vibrations, regardless of rotor balance, which must be isolated from the structure. The vibration isolators chosen were polyurethane bushing mounted on studs, which absorb the high frequency vibrations while transmitting the CMG loads. Figure 3.21 shows the vibration isolators installed at the base of the gimbal support bracket. Each isolator contains a threaded stud to adjust the compression on the bushings and change their absorption characteristics. Additionally, the bushings are a standard size and available in a variety of stiffnesses. Initial testing shows that the CMG vibrations have no measurable impact on the LN-200 angular rate or acceleration measurements.

*3.5.8 Gimbal Angle Alignment.* The G0AMH encoder provides the MicroAuto-Box with accurate measurements of the gimbal angle  $\delta$ , but require specifying the initial offset. By design, upper surface of the gimbal housing is parallel to the rotor axis, therefore aligning this surface parallel to the deck of SimSat aligns the angular momentum vector parallel to the deck and specified gimbal angle  $\delta$  of  $0^\circ$  for each rotor. The first step to aligning the gimbal was to level SimSat's deck using a bubble level. Next, a bubble level was placed on the gimbal assembly, and then rotated until the upper surface was parallel to the deck, shown in Fig. 3.22. Because SimSat's deck has a small amount of warping, the deck was leveled before adjusting each gimbal.



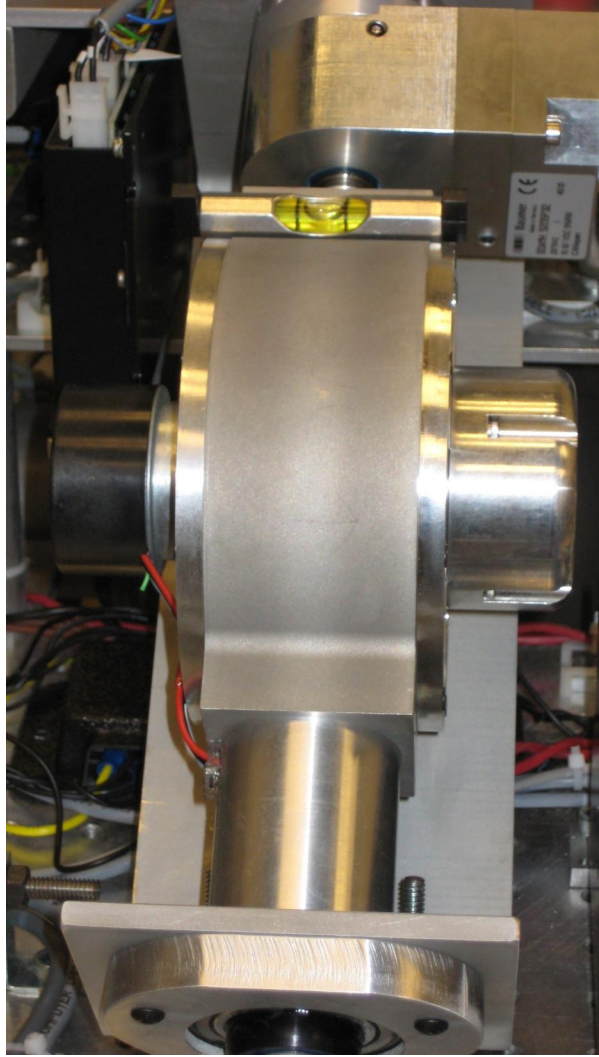


Figure 3.22: Gimbal Angle Alignment

### ***3.6 SimSat Control Program***

As stated in Section 3.3.3, the SIMULINK model executed by the MicroAutoBox and is responsible for all of the MicroAutoBox's command and data handling operations. Figure 3.23 shows the root level of the SIMULINK model used in this research, which consists of over 80 unique subsystems used to control every aspect SimSat's operation. Figure 3.23 was shown earlier as Fig. 3.10. The following sections will cover the subsystems that are most important to the operation of SimSat.

#### *3.6.1 Data Collection from Sensors.*





*3.6.1.1 LN-200.* The LN-200 subroutines were sourced from the work of Hines, McFarland, and Snider. These subroutines were originally written by Hines and include McFarland’s filters to minimize the impacts of “isolated gyro corruption” and Snider’s work to minimize impact of measurement bias.[11, 21, 34] The LN-200 subroutine was modified in this research to access the accelerometer data, and filtering was added to compensate for the noise in the accelerometer measurement. The accelerometer information was used as part of the State Update subroutines, covered in Section 3.6.2.

*3.6.1.2 CANOpen Devices.* All of the EPOS motor controllers used to command the fan/thrusters, reaction wheels, and CMG gimbals, as well as the G0AMH shaft encoders used to measure CMG gimbal angles, communicate with the MicroAuto-Box using the CANOpen protocol. In previous research efforts with SimSat, significant control issues existed due to the use of service data object (SDO) protocols for all communications. The solution to the communications issue was twofold. First, all process related communications were switched from SDO to process data object (PDO) protocols as discussed in Section 3.3.2. Second, all communications with the EPOS units were synchronized using a time division multiple access (TDMA) subroutine operating at 10 Hz and synchronized to the master clock, providing 100 separate communication windows. The EPOS units transmit only when they receive a remote transmission request (RTR), and each actuator set has its RTRs triggered via a TDMA block with a specific window number. Figure 24(a) shows the subroutine used to measure the status information from the CMG gimbal motor controllers. The reaction wheel and fan/thruster reads are similar. Port 1 is connected to an external TDMA trigger to maintain timing. Each EPOS unit provides the following information:

- Commanded angular velocity in rpm
- Actual angular velocity measured in rpm
- Status flag byte
- Mode of operation
- Current draw in mA

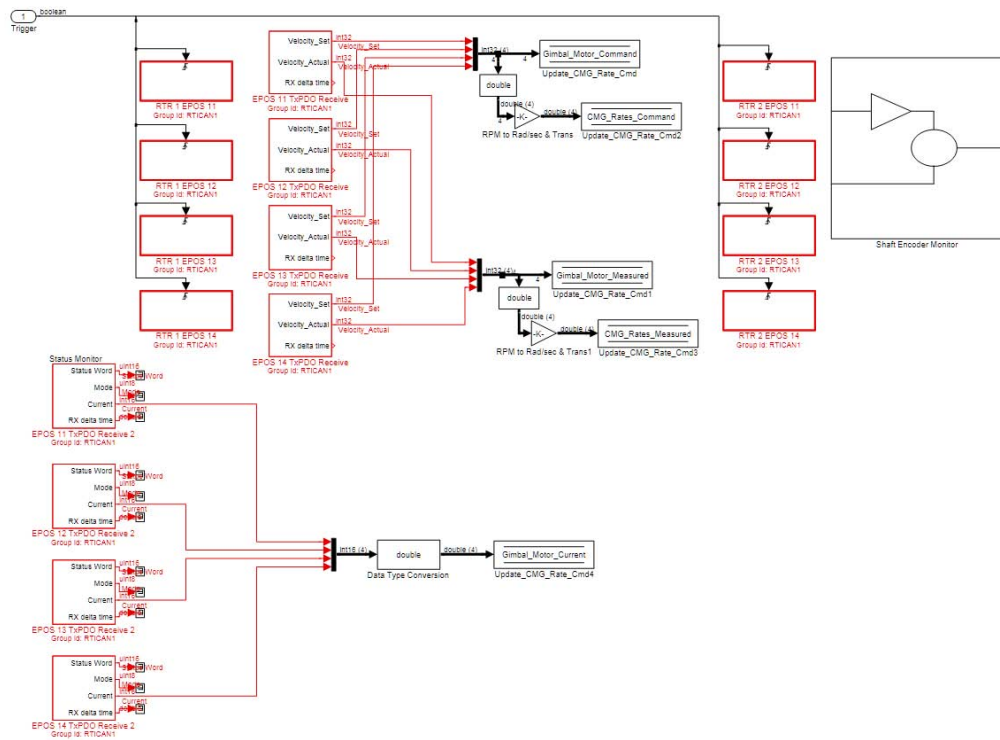
The information is provided to the operator via ControlDesk and to other subroutines within the SIMULINK program as needed.

The G0AMH shaft encoders do not include the RTR protocols, and instead transmit on a fixed interval. A subroutine was written that uses SDO protocols to alter the sample time, currently specified at 5 ms, and save the new to the internal flash memory so that the G0AMH maintain the specified configuration after a reboot. Figure 24(b) shows the subroutine uses to receive data transmitted by the encoders, as well as provide initial processing. Because the data does not arrive every cycle, SIMULINK's numeric derivative function could not be used to compute gimbal rates. Instead, the numeric derivative is calculated every time new encoder information is received using the measurements and timing data from the PDO receive block. The shaft encoder read function also applies the necessary conversion factors and gimbal offsets so the resulting measurements are usable within the SIMULINK program.

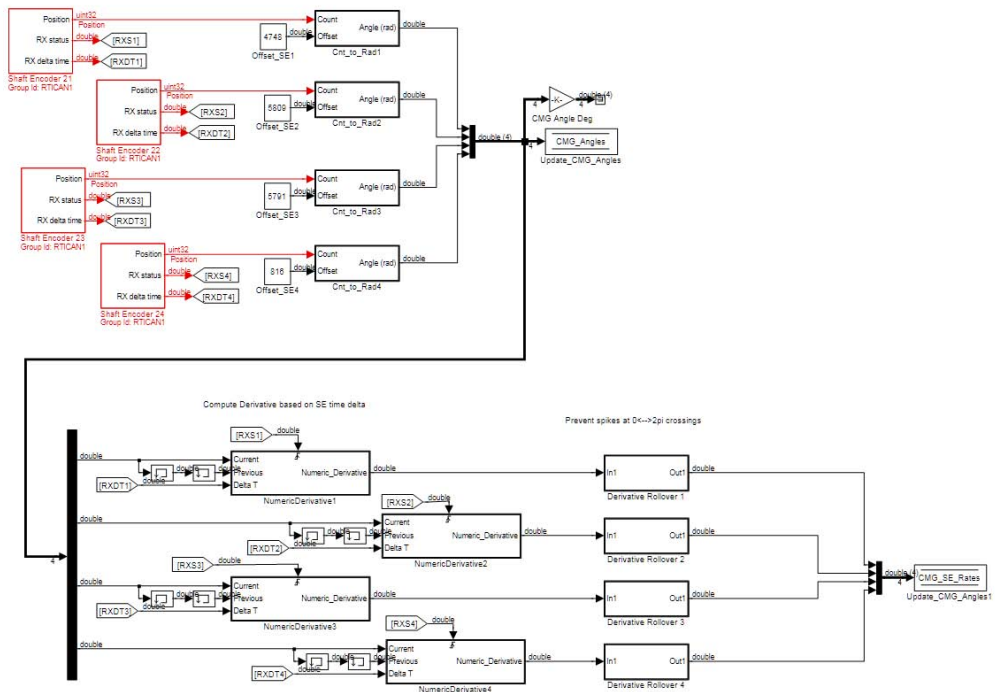
*3.6.2 Update State Information.* There are four key pieces of state information that must be maintained in order for the control system to function:

- Vehicle Rate
- Vehicle Orientation
- Reaction Wheel Angular Momentum
- CMG Array Angular Momentum

The LN-200 measures the vehicle rate directly, but does not measure the vehicle attitude. When SimSat is stationary, the LN-200's accelerometers provide a measurement of gravitational acceleration acting on SimSat about the body axes. The gravitational acceleration is constant in the inertial space, thus using the accelerometer data the orientation of SimSat's  $XY$ -plane relative to the inertial  $XY$ -plane can be computed. The  $XY$ -plane orientation information is used as the initial condition for the vehicle orientation. Centripetal accelerations are generated when SimSat is rotating because the LN-200 is not located at the center of rotation, and bias the acceleration measurement. The centripetal acceleration could be accounted for using the LN-200's body rate mea-



(a) CMG Gimbal Motor Receive



(b) CMG G0AMH Shaft Encoder Receive

Figure 3.24: CMG Receive Subroutines

surements but was determined to be unnecessary as the accelerometer was only used to determine an initial orientation when the vehicle was stable and not rotating.

The vehicle orientation state is maintained internally with quaternions using the subroutine shown in Fig. 3.25. Using quaternions ensures that SimSat can never lose its orientation state due to singularities in the kinematics because quaternions, by design, do not have singularities. The LN-200 body rate information is first converted into quaternion rates using the kinematics equations from Section 2.3.1, and then integrated using SIMULINK's discrete time integration subroutines operating at 1000 Hz. The initial orientation is either specified from the accelerometer measurements, or assumed to be aligned with inertial space and set to  $[0,0,0,1]$ .

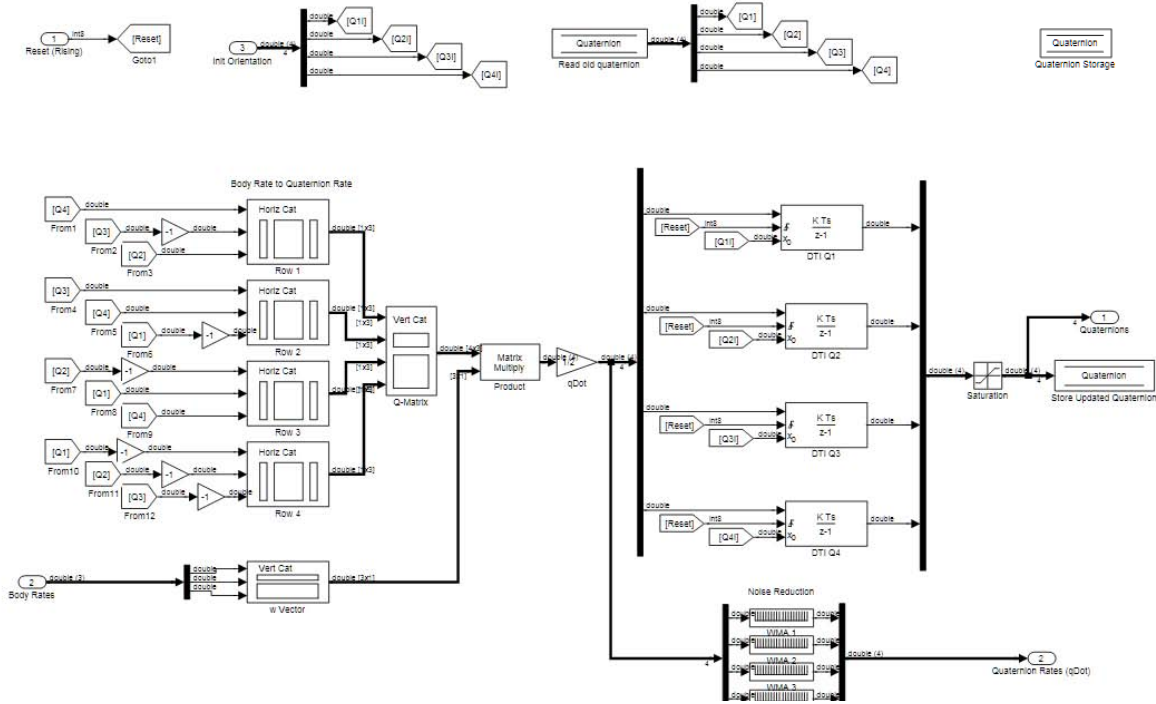


Figure 3.25: Vehicle Orientation Update

Reaction wheel angular momentum is maintained by the reaction wheel actuator subroutine, and calculated based on measurements provided by the EPOS unit of actual reaction wheel speed. The CMG array angular momentum is maintained by the CMG momentum calculation function, shown in Fig. 3.26 that computes the CMG momentum using shaft encoder measurements.

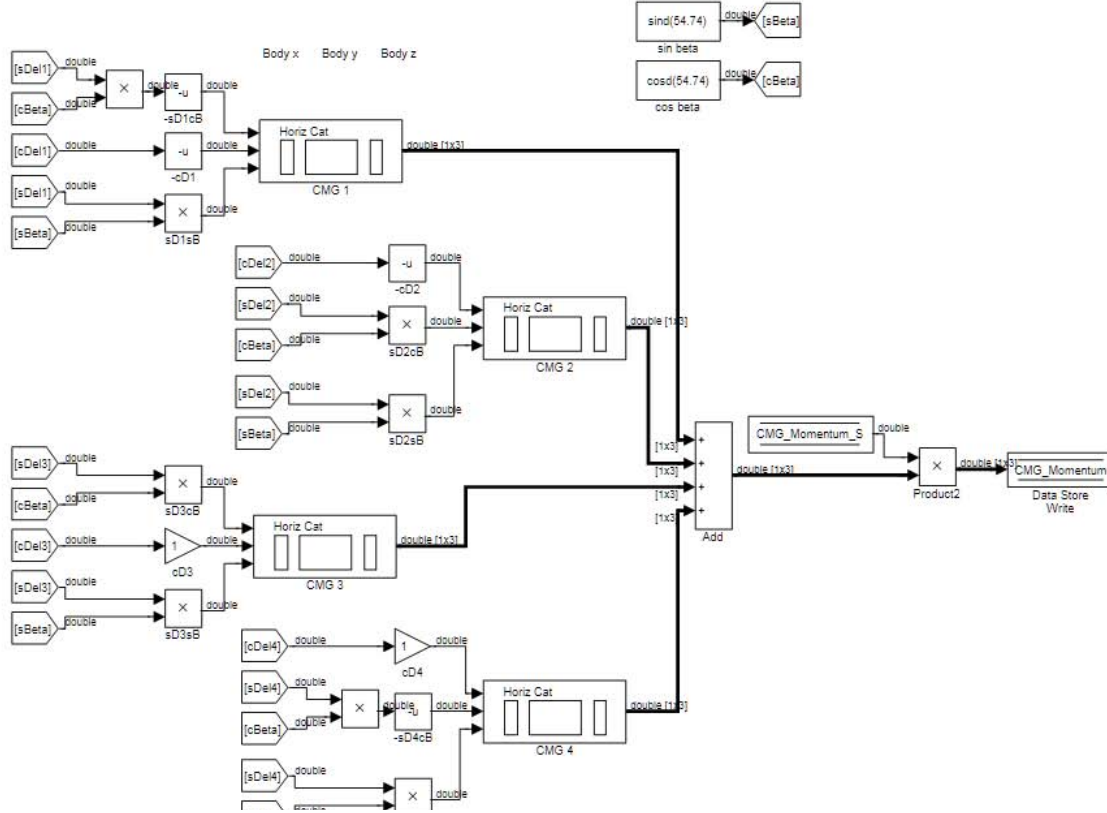
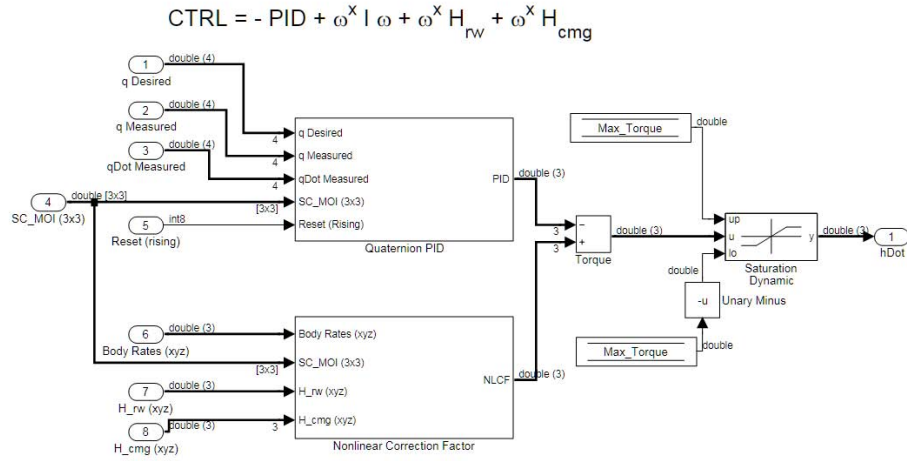
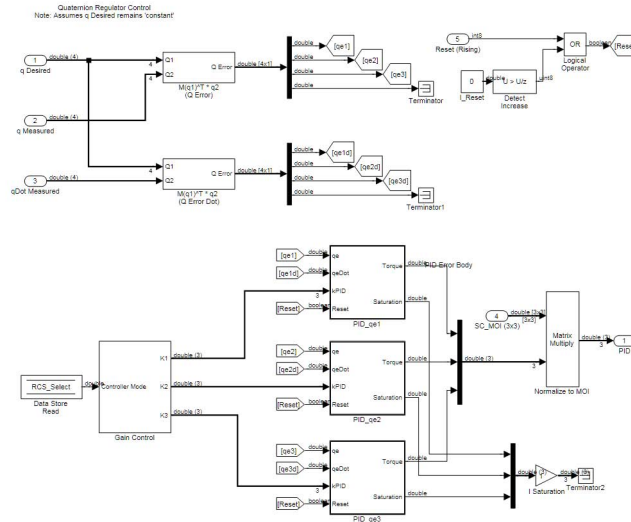


Figure 3.26: CMG Array Angular Momentum Update

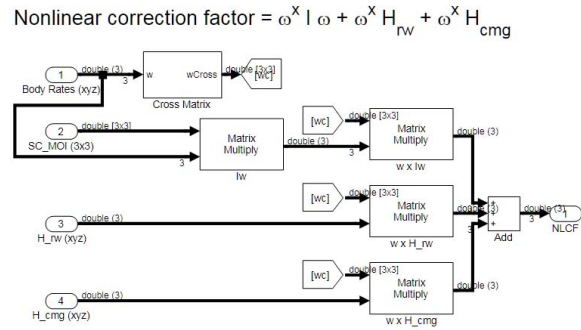
*3.6.3 Compute Control Solution.* SimSat uses the linearized PID controller derived in Section 2.5, shown in Figs. 3.27 (a), (b), and (c). The state information is provided by the various state update subroutines, and the target information is provided by through the ControlDesk or mLib interfaces. Proportional and integral control elements are calculated using the error between current and desired attitude. The integrator functions have saturation limits of  $\pm 0.5$  to limit integrator windup. In order to avoid noise amplification, the derivative control element is calculated based on the quaternion rate measured from the LN-200. The control subroutine calculates the desired change in angular momentum necessary to achieve the desired orientation. The control subsystem output was limited to  $\pm 0.25$  N-m commanded torque to prevent producing excessive control outputs. Because the PID operates on the quaternion error, the controller is self-re-linearizing about its current orientation and can handle discrete orientation changes up to  $35^\circ$  without encountering nonlinearities.



(a) Linearized Controller



(b) PID Subroutine



(c) Nonlinear Correction

Figure 3.27: Linearized PID Controller

*3.6.4 Generate Actuator Commands.* The actuator commands for the reaction wheels and CMG were implemented using the equations derived in Sections 2.3.4 and 2.4.

*3.6.4.1 Fan/Thruster Actuators.* The fan/thruster subsystem was sourced from the work done by McFarland, who noted that the fan/thruster system displayed nonlinear behavior.[21] The most accurate solution was to use a piecewise function to convert motor rates to torque, implemented on SimSat via a series of lookup tables. Due to the adjustment in fan location necessary to fit the larger reaction wheels, each fan/thruster's moment arm increased from 54.3 cm to 66 cm, or about 24%. A fixed gain of 0.8 was applied to fan torque commands to account for the change in moment arm. Because the fan/thruster actuators were only used for momentum dumping and experiment setup, not for data collection, a simple gain adjustment was deemed sufficient.

*3.6.4.2 Reaction Wheel Actuators.* The reaction wheel subsystem implements the equations derived in Section 2.3.4 using discrete time integration to calculate the desired reaction wheel rates based on the torque commanded by the controller. The integrator is reset when the system is shut down to prevent anomalous behavior upon restart, and has saturation limits to prevent overdriving the reaction wheel motors. The reaction wheel subroutine also implements a controlled spin-down function to prevent large torques from being applied to the vehicle when SimSat is disabled. Additionally, due to electrical current limitations the reaction wheels are limited to  $\pm 0.25$  N-m of torque effect and 300 rad/sec maximum rotor rate. Preliminary testing on a single reaction wheel indicated that these limitations could be relaxed if the current limiting circuitry is modified, as discussed in Section 5.2.1; however, the combined draw of three reaction wheels required the reaction wheel torque and angular rates be limited.

*3.6.4.3 CMG Moore-Penrose Pseudoinverse Steering Law.* SIMULINK's Signals Toolbox contains the Moore-Penrose pseudoinverse as a pre-defined block, and was used to provide the Moore-Penrose Pseudoinverse Steering Law (MPPSL). The Jacobian  $\mathbf{J}_H$ , derived in Section 2.3.5, is generated from the shaft encoder measurements,



processed via the SIMULINK pseudoinverse block, then multiplied by the single CMG angular momentum scalar and the desired change in angular momentum  $\dot{\vec{H}}$  to calculate the gimbal rates  $\bar{\delta}$ . The Moore-Penrose block produces large gimbal rate commands as the Jacobian approaches a singular state and the solution becomes numerically unstable.

**3.6.4.4 CMG Generalized Inverse Steering Law.** The Generalized Inverse Steering Law (GISL) was implemented in SIMULINK using the LU Inverse block of the Signals Toolbox, as shown in Fig. 3.28. The Jacobian  $\mathbf{J}_H$  and  $\mathbf{D}$  matrices derived in Section 2.4.3, in the code labeled as ‘A’ and ‘B’ respectively, are generated using the shaft encoder measurements, and used to calculate the generalized inverse. The generalized inverse is multiplied by the single CMG angular momentum scalar and the desired change in angular momentum  $\dot{\vec{H}}$  to calculate the gimbal rates  $\bar{\delta}$ , completing the GISL calculation. Like the MPPSL, the GISL will produce large gimbal rates when the matrix  $\mathbf{J}_H(\mathbf{J}_H + \mathbf{D})^T$  becomes singular which causes the solution to become numerically unstable.

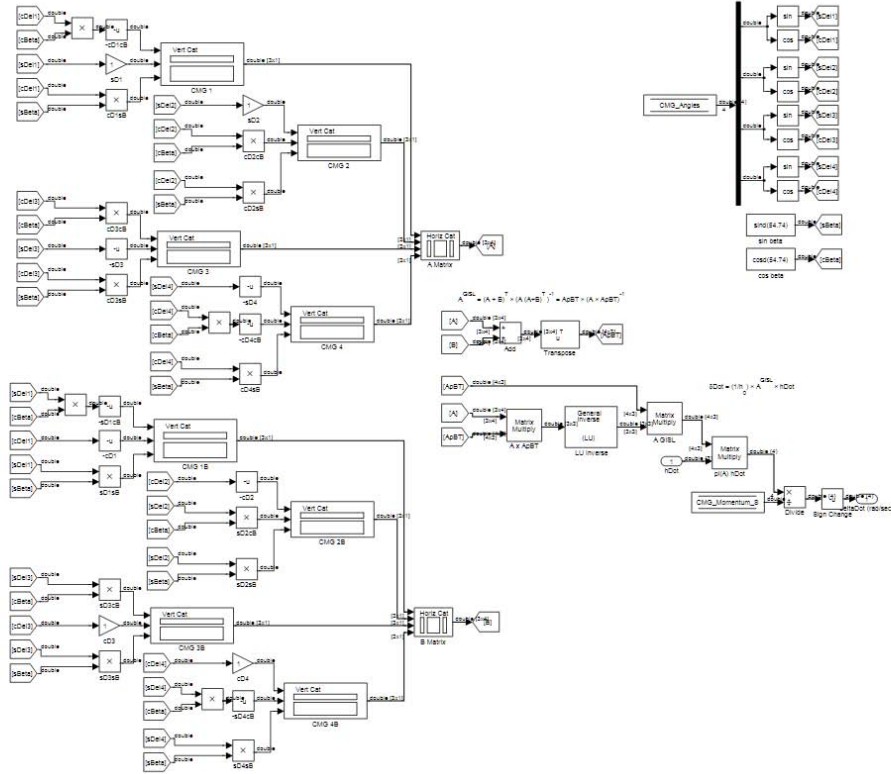


Figure 3.28: GISL Subroutine

*3.6.4.5 CMG Null Motion.* The null motion subroutine calculates the null motion solution by computing the null space of the Jacobian using the QR factorization block in the Signals Toolbox and a row reduction subroutine. The null space solution is then multiplied by a 4 second square wave to provide alternating clockwise and counterclockwise gimbal rates within the null space.

*3.6.5 Transmit Actuator Commands.* The EPOS motor control units are commanded via PDO protocols. As with the data reads, commands are synchronized using a TDMA subroutine. Additionally, each fan pair is tied together so that they operate synchronously, even if there is a communications delay, which minimizes transients at fan shutdown.

### **3.7 Vehicle Diagnostics**

Several tests were required to identify the system characteristics and validate the performance of SimSat using the both the reaction wheel ACS and CMG ACS. This section details the tests conducted on SimSat to determine the system parameters and validate the hardware.

*3.7.1 Vehicle Balancing.* In order to conduct any torque sensitive tests on SimSat, the center of mass and center of rotation must be aligned, minimizing the gravity disturbance torque so that the SimSat is neutrally stable, or balanced. Movement of components, such as wiring, causes small shifts in the center of mass during operation; thus removing the gravity disturbance torque completely was deemed impossible with the current hardware setup. Contact with the air-bearing pedestal almost always resulted in the center of mass shifting and would require a re-balance before tests could proceed. The balancing process took place in multiple stages. The first stage, coarse balancing, reduced the gravity disturbance torques until they were within SimSat's control envelope. Coarse balancing involved adding mass around the vehicle to move the center of mass. Most of this mass was added below the main deck on adjustable rods to offset the addition of the CMG array's mass above the main deck, shown in Fig. 3.29. Coarse adjustments were verified by having the vehicle balance unassisted for approximately 3 seconds.

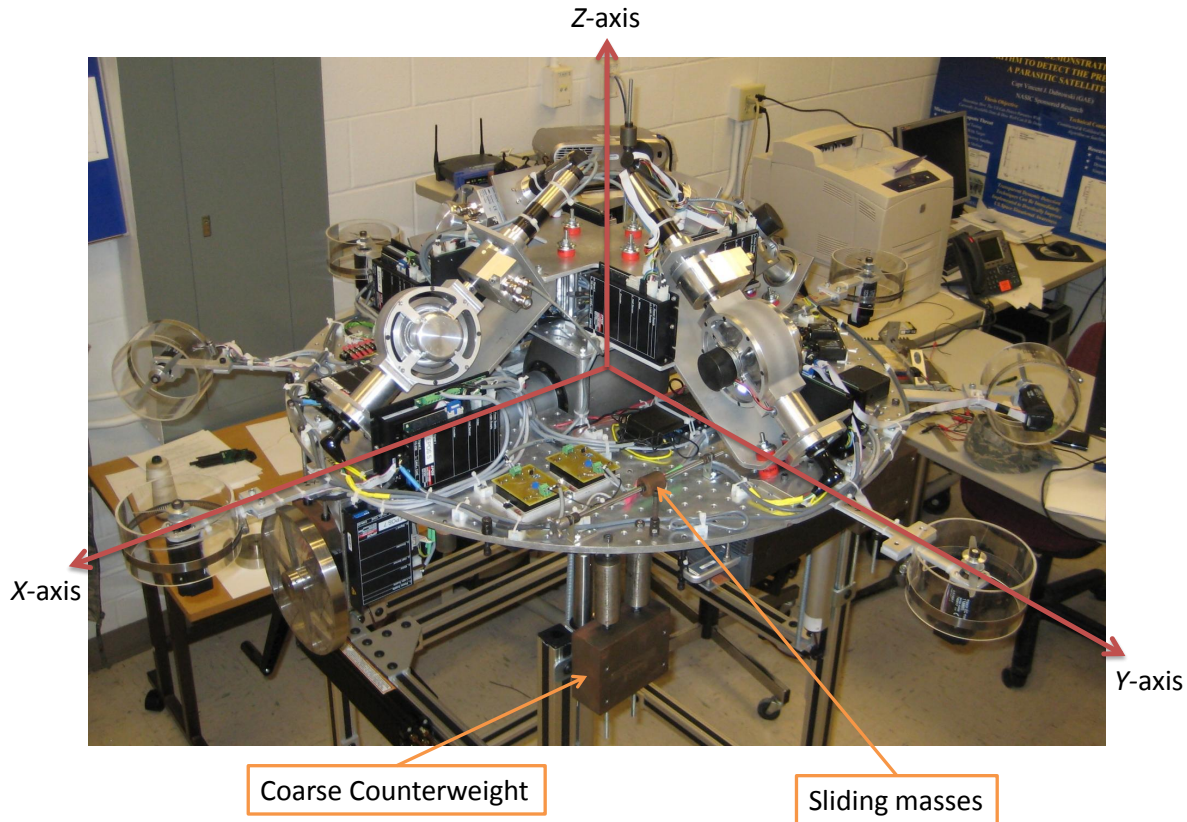


Figure 3.29: SimSat Body Axes and Counterweights

The second stage of balancing was also a coarse balancing, but using SimSat's ACS to measure the system response. The fan/thruster actuators were used to counteract the disturbance torques. At this stage of the research, the fan/thruster system had not been re-calibrated, nor the control gains tuned for optimal response to account for change in actuator location. The calibration tables and control gains from the previous SimSat configurations were used, which was deemed acceptable for this application because the goal was to reduce the control torque to zero, which was not affected by the change in actuator location. First, the  $Z$ -axis balance was intentionally set high by adjusting the location of the  $Z$ -axis counterweights, to make the vehicle unstable in the  $XY$ -plane and very sensitive to any  $XY$  imbalance. The vehicle was then brought to level using gravitational measurements from the accelerometers. Next, small masses, were placed or moved around the main deck until the indicated control torques were under 0.01 N-m. With the  $XY$ -plane balanced, the vehicle was tilted  $+10^\circ$  about the  $X$ -axis, and the

Z-axis weights were adjusted until the indicated control torques were under 0.01 N-m, indicating neutral stability. The vehicle was then tilted to  $-10^\circ$  about the X-axis to verify neutral stability. Tilt tests were also performed about the Y-axis.

The final stage of balancing was a fine balance, also using SimSat's ACS, but this time adjustments were performed using the 3 sliding masses mounted on the vehicle. There were two versions of this process performed, one using the fan/thrusters and one using the reaction wheels. The fan/thruster method was used prior to tuning the ACS gains. The first step was to balance SimSat using the previous technique of balancing the XY-plane, then the Z-axis. Instead of adding mass, the position of the 3 mass sliders was changed. Once the indicated control torques averaged less than 0.005 N-m, the fan/thruster system was disabled, allowing any disturbance torques to accumulate until noticeable motion occurred. The fan/thruster system was enabled, adjustments were made, and the process was repeated until neutral stability could be maintained for approximately 15 seconds.

The reaction wheel method for fine balancing was used once the reaction wheel ACS had been calibrated. The first step in reaction wheel balancing is to bring the vehicle to desired starting orientation using the fan/thruster actuators and the accelerometer to measure gravity level. During the alignment process, the reaction wheels are brought to a complete stop. Next, the ACS is switched into reaction wheel mode for 30 seconds, causing the reaction wheels to accumulate momentum to counteract the disturbance torque. After 30 seconds, the results are displayed and the fan/thruster system dumps the momentum. Adjustments are made to the slide balances until the reaction wheels accumulate less than 0.008 N-m-s of angular momentum, indicated average disturbance torque of 0.0025 N-m over the 30 second test.

*3.7.2 Reaction Wheel Calibration and Tuning.* As discussed in Section 3.4, the change from a 10 cm rotor to a 20 cm rotor altered the performance of the characteristics of the reaction wheel actuators. As a result, the reaction wheel subsystem required adjustment to ensure the EPOS velocity controller gains provided the optimum performance. The EPOS velocity controller is a PI controller, a subset of the PID

controller previously discussed. In the process of developing the 10 cm reaction wheel actuators, Snider, with the help of Maxon Motors, determined that the EPOS UI's automatic tuning algorithm was not designed to work with inertial load of the 10 cm reaction wheels.[34] Instead, tuning was performed manually, using Snider's gains of  $K_P = 15,000$  and  $K_I = 10$  as the initial value. The reaction wheel tuning test was also the final verification for the PDO and TDMA communications implementation. For the purposes of tuning, the TDMA algorithm for reaction wheel measurements was set to 100 Hz sampling, while the reaction wheel commanding was maintained at 10 Hz. A series of  $\pm 0.25$  N-m torque commands were issued to the reaction wheel algorithm, which then issued a ramp command the EPOS controller. The gains  $K_P$  and  $K_I$  were increased until the rise time, settling time, and overshoot were acceptable. Due to current draw limitations specified in Section 3.4, a small rise time lag was accepted. The reaction wheel saturation limits were also modified during this time to prevent exceeding the electrical current limitations. The maximum speed of the EC-45 motor is limited by the system voltage at 1030 rad/sec, but the motor's electrical current consumption behaves according to

$$I = \frac{\tau \Psi}{\eta(\Psi) V} \quad (3.1)$$

where  $I$  is the motor current required,  $\tau$  is the torque generated (including torque to overcome losses),  $\Psi$  is the instantaneous operating speed,  $\eta(\Psi)$  is the motor efficiency which is a function of  $\Psi$  and has a maximum value of 85% for the EC-45, and  $V$  is the supply voltage. In order to ensure that 0.25 N-m of torque is available for all reaction wheels under all operating conditions, the reaction wheels require a saturation limit lower than the 1030 rad/sec that is theoretically available. The results of the reaction wheel tuning are in Section 4.2, with the final gains being  $K_P = 20,000$  and  $K_I = 200$  for all three reaction wheels. These gains remained constant for all other tests conducted in this research. [34, 17, 18].

*3.7.3 Vehicle MOI Testing.* SimSat's MOI affects all aspects of the vehicle's performance, and must be known in order to verify that the CMG and reaction wheel

subsystems are performing as designed. SimSat is both too large and too heavy to test on AFIT's XLR250 test stand. In previous work with SimSat, the vehicle MOI was measured by applying a reference torque to the vehicle and measuring its angular acceleration.[21, 34] According to Euler's equation, previously derived in Section 2.3.2 and restated here as Eq (3.2)

$$\vec{M} = \mathbf{I} \dot{\vec{\omega}}_{bi} + \vec{\omega}_{bi} \times \mathbf{I} \vec{\omega}_{bi}. \quad (3.2)$$

In order to remove the nonlinear term  $\vec{\omega}_{bi} \times \mathbf{I} \vec{\omega}_{bi}$ , the control torque was assumed to be applied to a single principal axis, which simplifies Eq. (3.2) to

$$\begin{aligned} M_1 &= I_1 \dot{\omega}_1 \\ M_2 &= I_2 \dot{\omega}_2 \\ M_3 &= I_3 \dot{\omega}_3 \end{aligned} \quad (3.3)$$

where the 1-, 2-, and 3-axes are the X-, Y-, and Z-axes, respectively. In previous research, the torque was applied by adding a known mass at a fixed distance. While the previously used method produced acceptable results, the manual effort required precluded taking a large number of samples. Instead, a new method was developed in this research that uses the reaction wheel subsystem to compute the vehicle's MOI using Euler's equation with the reaction wheels, derived in Section 2.3.4 and restated for convenience here

$$\vec{M} = \mathbf{I}_b \dot{\vec{\omega}}_{bi} + \begin{bmatrix} I_{rw} \dot{\Psi}_1 \\ I_{rw} \dot{\Psi}_2 \\ I_{rw} \dot{\Psi}_3 \end{bmatrix} + \vec{\omega}_{bi} \times \left( \mathbf{I}_b \vec{\omega}_{bi} + \begin{bmatrix} I_{rw} \Psi_1 \\ I_{rw} \Psi_2 \\ I_{rw} \Psi_3 \end{bmatrix} \right). \quad (3.4)$$

Assuming the external torques are negligible and angular motion is constrained to the principal axis, Eq. (3.4) reduces to

$$\vec{0} = \mathbf{I}_b \dot{\vec{\omega}}_{bi} + \begin{bmatrix} I_{rw} \dot{\Psi}_1 \\ I_{rw} \dot{\Psi}_2 \\ I_{rw} \dot{\Psi}_3 \end{bmatrix} \quad (3.5)$$

which can be decoupled and re-arranged to yield

$$\begin{aligned} I_1\dot{\omega}_1 &= -I_{rw}\dot{\Psi}_1 \\ I_2\dot{\omega}_2 &= -I_{rw}\dot{\Psi}_2 \\ I_3\dot{\omega}_3 &= -I_{rw}\dot{\Psi}_3 \end{aligned} \tag{3.6}$$

It is important to note that Eq. (3.6) is only valid if motion occurs about a single axis. Using Eq. (3.6), SimSat's scalar moments of inertia can be measured by applying a fixed angular acceleration to the reaction wheel and measuring SimSat's response using the LN-200. The advantage to using the reaction wheels is to negate the cosine losses encountered when using an applied mass to generate a gravitational torque as explained by McFarland.[21] Additionally, the reaction wheel technique requires no operator interaction, a potential source of error and can be easily repeated. Using MATLAB and mLib, a subroutine was written to perform the following actions:

1. Establish starting orientation and null out the angular rates using fan/thruster subsystem
2. Disable fan/thruster subsystem
3. Begin recording of body rates  $\vec{\omega}$  and reaction wheel rates  $\vec{\Psi}$
4. Wait 1 sec for any disturbances to accumulate. If so, the test can be rejected in post processing.
5. Apply fixed angular acceleration corresponding to 0.1 N-m to specified reaction wheel for 4 sec
6. Terminate recording
7. Zero reaction wheel rates
8. Enable fan thruster subsystem and re-establish initial orientation

Prior to the test, the vehicle was fine balanced using the fan balance technique. The MOI test was performed twenty times in both the positive and negative directions for each axis. For the  $X$ - and  $Y$ -axes, the starting orientation was  $\pm 5^\circ$ , depending on direction,

to provide sufficient travel to prevent collision with the pedestal. Tests about the  $Z$ -axis were conducted from level orientation. The results of the MOI test are discussed in Section 4.3.

*3.7.4 CMG Rotor and Gimbal Testing.* Before testing the CMG array's performance as an attitude control actuator, it was necessary to perform a component level validation and calibration. The components tested were the rotor motors, gimbal motors, and gimbal shaft encoders to ensure each component was functioning as intended and eliminate it as a source of error.



Figure 3.30: CMG Rotor Calibration

Each CMG rotor is driven by a Maxon EC-45 flat motor with integrated control electronics. The control electronics drive the motor to a fixed speed based on the supplied signal voltage, with an ideal no-load linear response of 60 rpm/V up to a maximum of 6000 rpm at 10 V. Each rotor motor is connected to the MicroAutoBox digital to analog converter (DAC), which can produce a voltage ranging from 0-4.5 V in increments of 0.0044 V, allowing the MicroAutoBox to specify speeds up to 2700 rpm, slightly below the design speed of 3000 rpm. A Cole-Parmer 8210 photo-tachometer, shown in Fig. 3.30 was used to measure the rotation rate of each rotor and verify the performance. Due to bearing friction, each rotor had a unique speed, ranging from 2630 rpm to 2690 rpm for a 4.5 V signal. While the difference was less than 2%, it was significant enough



to require adjustment. Therefore, each rotor speed was adjusted downward to achieve a measured rate of 2600 rpm, verified using the photo-tachometer to  $\pm 10$  rpm. These voltage offsets were hard coded into the SIMULINK model. Operating at 2600 rpm, the rotors store approximately 0.45 N-m-s of angular momentum each, giving the CMG array approximately  $\pm 1.47$  N-m-s about the Z-axis or  $\pm 1.42$  N-m-s about the X- and Y-axes, a reduction of about 13% from the original design, but still within the design specifications.

Each CMG gimbal motor is controlled by an EPOS 70/10 module. The EPOS UI's automatic tuning program was used to determine the PI controller gains. Unlike the reaction wheels, the CMG gimbal mechanism has a low enough moment of inertial relative to the motor, that the automatic tuning algorithm was able to specify gains which provided acceptable rise and settling times with minimal overshoot. With the gimbal motors gains calibrated, it was necessary to verify that the gimbal motors and gimbal shaft encoders were producing the same measurements. The test consisted of applying a fixed rate to the gimbal motor and validating that the shaft encoder position measurements matched the expected rate. Additionally, shaft encoder measurements were differentiated in real-time using information from the PDO communications block, discussed in Section 3.6. The tests indicated solid correlation between shaft encoder measurements and gimbal motor commands. Full results of these tests are provided in Section 4.4.

*3.7.5 CMG and Reaction Wheel Gain Tuning.* The PID attitude controller gains were adjusted manually using the modified Ziegler-Nichols tuning rules.[24] A  $20^\circ$  Z-axis slew was used for baseline calibration of each actuator system, with the focus on minimizing settling time and steady state error. The fan/thruster gains were tuned first to allow the fan/thruster system to be used for stabilization and momentum dumping between experimental runs. The initial fan/thruster gain values were taken from McFarland's PID setup used for system calibration. The  $K_P$  was increased until rise time reached an acceptable value.  $K_D$  was then increased until settling time reached an acceptable level with minimal overshoot, maintaining an under-damped response.  $K_I$  was then increased to achieve the desired steady state response. Increasing  $K_I$  ad-

versely affected the settling time, therefore  $K_P$  and  $K_D$  were adjusted to compensate for changes in  $K_I$ . Lastly, the  $Z$ -axis gains were tested for suitability about the  $X$ - and  $Y$ -axis. The final fan/thruster control gains were  $K_P = 5, K_I = 0.2$ , and  $K_D = 8$ . The fan/thruster gains were used as the initial values for tuning the reaction wheel and CMG systems. Additionally, the tuned fan/thruster system minimized setup time for repeat experiments.

The reaction wheel control gains were tuned using the same techniques used to tune the fan/thruster actuators, using a  $20^\circ$   $Z$ -axis slew as a split between the  $X$ - and  $Y$ -axis requirement and the  $Z$ -axis requirement. Initial attempts were hampered by a significant phase lag present in the EPOS motor controller. Tuning the reaction wheel controller's PI gains, addressed in Section 3.7.2 reduced the phase lag to 0.4 sec, but was unable to eliminate the phase lag entirely. As a result of the phase lag, the gains determined for the fan/thruster ACS caused an unstable system when used with the reaction wheel ACS.  $K_P$  was therefore reduced until stability was achieved, followed by adjusting  $K_D$  and  $K_I$  to achieve the desired settling time and steady state response. The final reaction wheel control gains chosen were  $K_P = 2, K_I = 0.1$ , and  $K_D = 4$ .

The initial CMG control gains were set using the fan/thruster values. The CMG array was well behaved with the fan/thruster values, such that any adjustment to  $K_P$  decreased overall performance. Increasing  $K_D$  did, however, improve settling time, as did decreasing  $K_I$  with minimal impact to the steady state performance of the ACS. The final values for the CMG gains were  $K_P = 5, K_I = 0.1$ , and  $K_D = 11$ .

*3.7.6 CMG and Reaction Wheel Rest-to-Rest Testing.* The positional accuracy and rest-to-rest, or slew, maneuver capabilities of the CMG and reaction wheel actuators was tested by commanding specific orientations to the vehicle, and recording the resulting vehicle response. The purpose of this test set was to test SimSat's ACS performance against the performance specifications presented in Chapter I. An mLib script was written to automate the following testing procedure:

1. Using the fan/thrusters, establish and maintain level alignment using the accelerometer to determine  $XY$ -plane level

2. Zero reaction wheel rates and align gimbals to starting angles
3. Simultaneously disable fan/thruster system and enable system under test
4. Command desired orientation
5. Record data for 30 seconds
6. Switch back to fan/thruster system upon test completion and hold position

The  $X$ - and  $Y$ -axis slew maneuvers were  $\pm 10^\circ$ , while the  $Z$ -axis slew maneuvers were  $\pm 30^\circ$ . Table 3.3 lists the actuator configurations tested. Two initial conditions were chosen for the CMG gimbals, both of which are zero net momentum configurations, to observe different singular configurations within the CMG momentum envelope. The specific conditions of  $\bar{\delta}_0 = [0, 0, 0, 0]$  and  $\bar{\delta}_0 = [\pi/2, -\pi/2, \pi/2, -\pi/2]$  were chosen because they are the two extreme starting conditions within the zero net momentum solution space.

Table 3.3: Rest-to-Rest Actuator Configurations

Actuator	Steering Law	Initial Gimbal Angles $\bar{\delta}_0$
Reaction Wheels	N/A	N/A
CMG	MPPSL	$[0, 0, 0, 0]$
CMG	MPPSL	$[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$
CMG	GISL	$[0, 0, 0, 0]$
CMG	GISL	$[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

Results and analysis of the rest-to-rest testing are Section 4.5.

*3.7.7 CMG and Reaction Wheel Torque Testing.* Both the CMG and reaction wheel ACS were tested for both torque generation capability and accuracy by applying a mass to the edge of the vehicle, 60 cm from the center of rotation, with the control system enabled. For the  $X$ - and  $Y$ -axis tests, a short piece of string was attached to the vehicle along one of the fan support arms at 60 cm from the center of SimSat's air-bearing. For the  $Z$ -axis, a string and low friction pulley were used, with pulley positioned so the string was in the  $XY$ -plane and tangent to the vehicle, ensuring only a  $Z$ -torque would be applied. Prior to each test set, the vehicle was fine balanced using the reaction wheels to

minimize gravitational disturbance torques and account for any disturbances introduced by the string. An mLib script was written to perform the following test procedure:

1. Using fan/thrusters, establish and maintain level alignment using the accelerometer to determine  $XY$ -plane level
2. Zero reaction wheel rates and align gimbals to starting angles
3. Request operator apply disturbance torque
4. Begin recording and wait 1 second.
5. Simultaneously disable fan/thruster system and enable system under test
6. Record data for specified time (test dependent)
7. Switch back to fan/thruster system upon test completion and hold position

Figures 3.31 (a) and (b) show the setup of the  $X$ - and  $Z$ -axis tests. Due to actuator locations, all torques were applied in the negative body axis directions. The same actuator configurations used for rest-to-rest maneuvers were used for disturbance torque testing, with three different torques being applied:

1. 0.101 N-m - 17.2 g
2. 0.199 N-m - 33.8 g
3. 0.250 N-m - 42.6 g

A no load control test was also performed using the reaction wheel ACS and the CMG ACS with the Moore-Penrose Pseudoinverse Steering Law. In order to ensure SimSat maintained a balanced configuration, tests were stopped whenever the control system saturated or reached an impassable singularity. This prevented SimSat from colliding with the pedestal which would alter the vehicle's balance as wiring shifts due to impact shock. Results and analysis of the torque testing are in Section 4.6.

*3.7.8 CMG Null Motion.* In order to verify the null motion behavior of the CMG array, the null motion gimbal rates were applied to the vehicle. Ideally, the null motion solution should not cause any angular momentum exchange with the rest of SimSat.

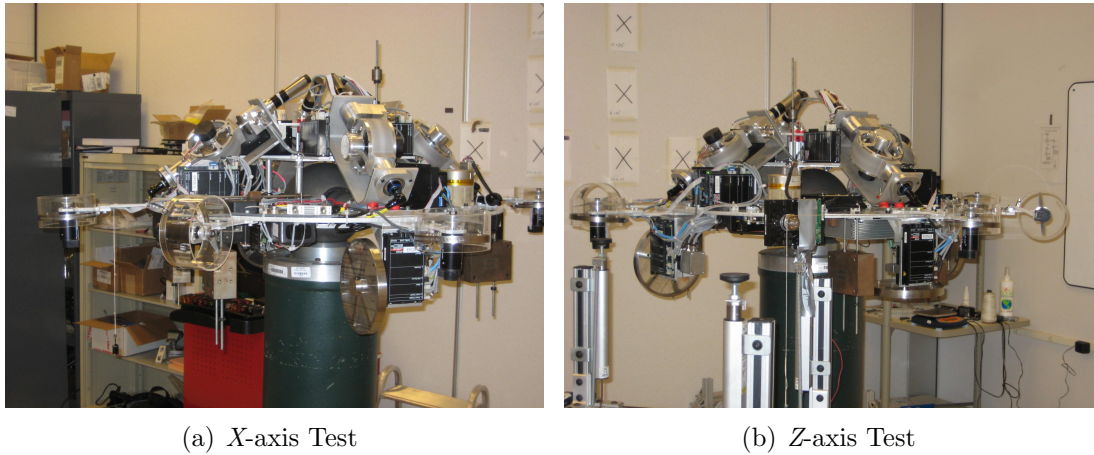


Figure 3.31: Disturbance Torque Testing

Initial null motion tests were done by applying the null motion solution, starting from the zero angular momentum gimbal configuration with vehicle at rest and the reaction wheel and fan/thruster ACS disabled. Despite performing a fine balance before the test, the disturbance torques could not be separated from momentum disturbances generated by the CMG movement. In order to isolate the effect of disturbance torques, the null motion behavior was applied while using the reaction wheel ACS. The reaction wheel ACS acted as the sensor by storing any angular momentum the CMG array exchanges with body. The testing procedure used was:

1. Using fan/thrusters, establish and maintain level alignment using the accelerometer to determine  $XY$ -plane level
2. Zero reaction wheel rates and align gimbals to starting angles
3. Switch to reaction wheel system
4. Record disturbances for 30 seconds
5. Switch to fan/thruster system
6. Using fan/thrusters, establish and maintain level alignment using the accelerometer to determine  $XY$ -plane level
7. Zero reaction wheel rates and align gimbals to starting angles
8. Switch to reaction wheel subsystem

9. Initiate CMG null motion behavior - 2 seconds clockwise, 2 seconds counterclockwise
10. Record disturbances for 30 seconds
11. Switch back to fan/thruster system upon test completion and hold position

By comparing the reaction wheel behavior with and without null motion, the majority of the disturbance torques can be isolated from any momentum exchange generated by the null motion. Results from the null motion test are in Section 4.7.

*3.7.9 CMG Torque Multiplication Testing.* In order to calculate the potential torque multiplication of the CMG array, it was necessary to measure the torque generated by the gimbal motors for a given gimbal rate, and the corresponding torque output of the CMG at a specific gimbal rate. Due to losses in both the bearing and transmission, the motor current usage cannot be used to directly measure the theoretical torque applied by motor to the gimbal. Instead, the reaction wheel system was used to indirectly measure the torque applied to the CMG gimbals using conservation of angular momentum. Prior to installation, the CMG gimbal assembly's gimbal axis scalar moment of inertia was determined to be  $3.52 \times 10^{-3} \text{ kg-m}^2$  using the XR250. Because of the pyramid configuration, applying an equal torque to all four gimbals will result in a  $Z$ -axis momentum exchange. With the rotors disabled, the torque applied by the gimbals, accelerating equally, is

$$T_G = \frac{1}{4 \cos(54.74^\circ)} T_{Ga} = -\frac{1}{4 \cos(54.74^\circ)} \dot{H}_z \quad (3.7)$$

where  $T_G$  is the torque applied to the gimbal,  $\cos(54.74^\circ)$  is the  $\beta$  angle, and  $T_{Ga}$  is the net torque effect, or change in angular momentum of the gimbal assembly on SimSat. With the reaction wheel ACS enabled, the change in angular momentum will be counteracted by the  $Z$ -axis reaction wheel, providing a measurement for  $\dot{H}_z$  in Eq. (3.7). In order to account for disturbance torques, a zero gimbal motion test was performed prior to the gimbal rate tests.

The second component to testing torque multiplication was to determine the torque generated by each CMG when a given gimbal rate was applied. With the pyramid

configuration, if all CMGs have the same initial angle and equal rate, then the torque applied will purely about the  $Z$ -axis, with peak torque magnitude occurring at  $\delta = 0$  and  $\delta = 180^\circ$ . In terms of the individual CMGs, the torque produced is

$$T_{cmg} = \frac{1}{4 \sin(54.74^\circ)} T_{cmga} = -\frac{1}{4 \sin(54.74^\circ)} \dot{H}_z \quad (3.8)$$

where  $T_{cmg}$  is torque effect generated by each individual CMG,  $T_{net}$  is the total torque effect produced, and  $\frac{1}{4 \cos(54.74^\circ)}$  accounts for geometry and cancellation within the CMG array. By applying a fixed gimbal rate while running the reaction wheel ACS, the torque generated by the CMG array is counteracted by the reaction wheels providing a measurement of  $\dot{H}_z$ . Measuring at the  $\delta = 0$  crossing will provide a measure of the peak torque produced by the CMG array.

Therefore, the following test procedure was used to determine the torque amplification:

1. Using fan/thrusters, establish and maintain level alignment using the accelerometer to determine  $XY$ -plane level
2. Zero reaction wheel rates and align gimbals to starting angles
3. Switch to reaction wheel system
4. Record disturbances for 30 seconds
5. Switch to fan/thruster system
6. Using fan/thrusters, establish and maintain level alignment using the accelerometer to determine  $XY$ -plane level
7. Zero reaction wheel rates and align gimbals to starting angles
8. Switch to reaction wheel system and apply fixed gimbal rates for 30 seconds with CMG rotors disabled
9. Record gimbal torques gimbal motor current draw
10. Switch to fan/thruster system

11. Using fan/thrusters, establish and maintain level alignment using the accelerometer to determine  $XY$ -plane level
12. Zero reaction wheel rates, align gimbals to starting angles, and enable CMG rotors
13. Switch to reaction wheel system and apply fixed gimbal rates for 30 seconds with CMG rotors enabled
14. Record CMG array torques
15. Switch to fan/thruster system upon test completion and hold position

The data collected was then post processed using Eqns. (3.7) and (3.8) to calculate torque multiplication. Additionally, the gimbal actuation torque was calculated using the Maxon motor specifications for the EC-MAX-30 and the recorded gimbal motor current draw which estimates the total torque produced by the motors including losses to the gimbal bearings. Results and analysis are in Section 4.8.

### ***3.8 Summary***

Chapter III presented the methodology used in the design, construction, integration, software development, and validation of the low cost CMG array for SimSat and revisions to the reaction wheel subsystem. The relevant hardware, existing and new, as well as software interfaces used during development were discussed. SimSat's operating program, including actuator communications, control law and steering law was presented. Lastly, the testing procedures used to characterize and validate SimSat's CMG and reaction wheel subsystems was described.



## IV. Results and Analysis

### 4.1 Introduction

Chapter IV evaluates the results and analysis of the system validation tests described in Section 3.7, specifically:

1. Reaction Wheel Calibration and Tuning
2. Vehicle MOI Test
3. CMG Gimbal Test
4. CMG and Reaction Wheel Rest-to-Rest Test
5. CMG and Reaction Wheel Torque Test
6. CMG Null Motion Test
7. CMG Torque Multiplication Test

As described in Section 3.7, all data was captured on the Mini-Box PC using mLib scripts to execute the experiment. Automating the experiments and data capture in scripts, the tests described in Section 3.7 can be easily repeated and evaluated for consistency. This chapter presents a subset of the results, along with the accompanying analysis of the complete data collected. Additional results figures are provided in Appendix A.

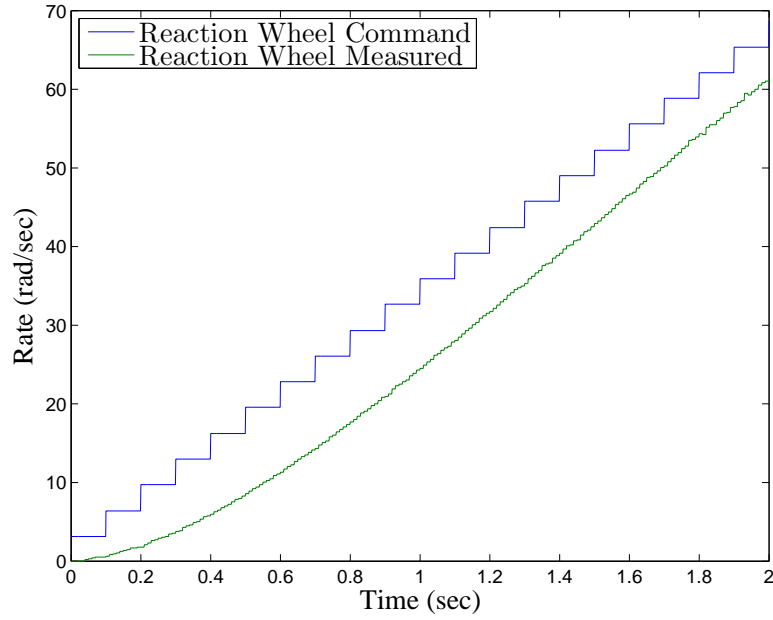
### 4.2 Reaction Wheel Calibration and Tuning Results

This section presents the results of the reaction wheel calibration and tuning tests described in Section 3.7.2; with the goal of validating the communications protocols and adjusting EPOS PI controller settings to account for the increased MOI of the 20 cm reaction wheels. The results of the  $X$ -axis, -0.25 N-m test are shown in Fig. 4.1, where Fig. 4.1(a) are the original EPOS PI gains of  $K_P = 15,000$  and  $K_I = 10$ , and Fig. 4.1(b) are the final gains selected. The  $Y$ - and  $Z$ -axis reaction wheels had nearly identical responses, and can be found in Appendix A1.1. It is important to note that for this test, the TDMA subroutine for measuring the reaction wheel was set to 100 Hz to observe the behavior between command intervals. The initial gain values of  $K_P = 15,000$  and  $K_I = 10$  resulted in significant phase lag, seen in Fig. 1(a) where the response fails

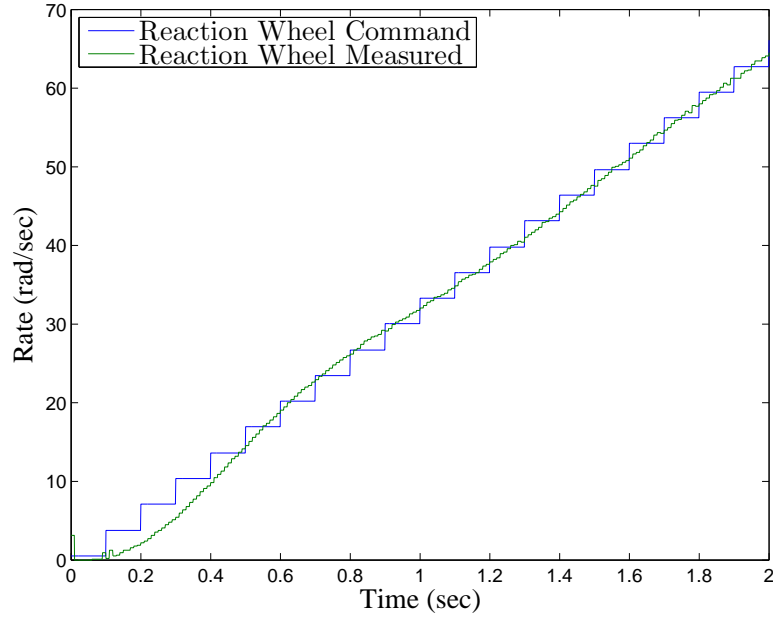
to match the commanded ramp function within the 2 second window. The impact of the phase lag was significant enough that initial attempts to tune the PID attitude controller for the reaction wheel system were completely unsuccessful because of phase lag induced instability. Increasing the gains to  $K_P = 20,000$  and  $K_I = 200$  reduced the phase lag to about 0.5 seconds, but was unable to eliminate it. Increasing the gains above  $K_P = 20,000$  and  $K_I = 200$  resulted in the EPOS controllers exceeding the available current which caused a low voltage fault and would shut down the EPOS. The  $2.7\ \Omega$  resistor bank nominally allows for 13.7 A at 37 V; however, the battery voltage varies from 40 V when fully charged to 34 V when nearly discharged. As the battery voltage drops, so does the current limit imposed by the resistor bank. Relaxing the current limitations by decreasing the total resistance of the resistor array should allow the reaction wheel controller PI gains to be increased and further reduce the phase lag in the reaction wheels.

During this controller tuning phase, the reaction wheel saturation limits were also locked down. Initially, a 630 rad/sec limit was imposed on the reaction wheels but this limit not strict enough and resulted in a low voltage fault on the reaction wheel EPOS units anytime more than one wheel approached saturation. When a low voltage fault occurs, the EPOS unit shuts down the motor power circuitry, allowing the reaction wheel to spin freely. The high gains in the PI controller cause the EPOS to draw significantly more current than is predicted by the motor power equation listed in Section 2.3.4. The digital shunt regulator's (DSR) capacitors address short term transients, but are insufficient to handle sustained loads. The solution chosen was to lower the saturation limit to 314 rad/sec, which eliminated the low-voltage under most operating conditions. The lowered saturation limit means that the 20 cm reaction wheels provide 2.49 N-m-s of angular momentum storage and still provide 0.25 N-m of torque, a 22% increase for the  $X$ - and  $Y$ -axis reaction wheels but a 37% decrease for the  $Z$ -axis reaction wheel. The 20 cm wheel is, however, roughly  $1/3^{\text{rd}}$  the mass of the 10 cm  $X$ - and  $Y$ - axis reaction wheels. Increasing the current limit by decreasing the resistance of the resistor array to  $1.6\ \Omega$  should be sufficient to raise the saturation limit to 500 rad/sec and allow the 20 cm

wheel to store the same angular momentum as the 10 cm  $Z$ -axis wheel. Modification of the resistor array is discussed further in Section 5.2.1.



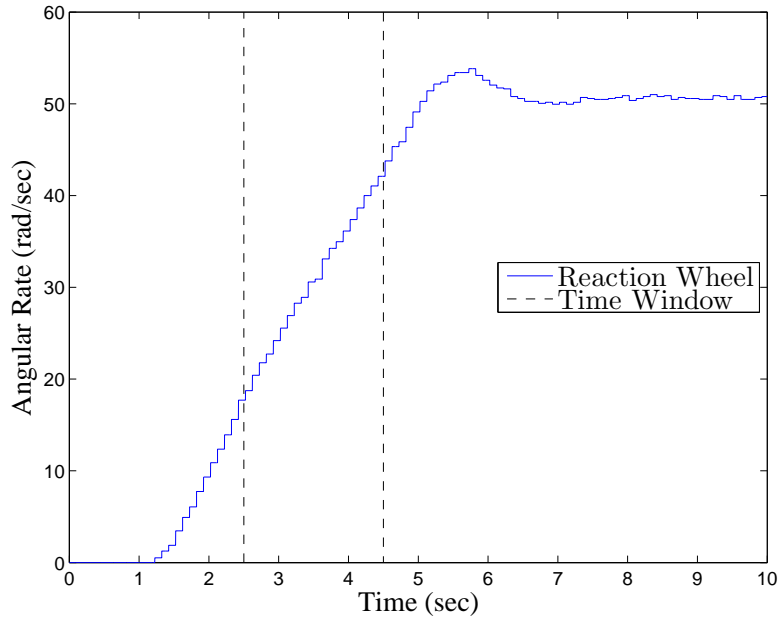
(a)  $K_P = 15,000$  &  $K_I = 10$



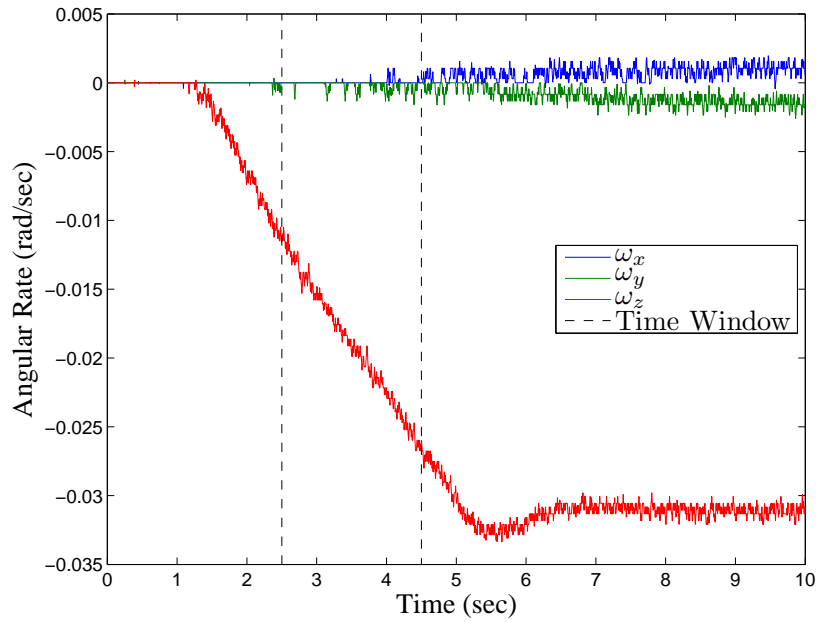
(b)  $K_P = 20,000$  &  $K_I = 200$

Figure 4.1: X-axis Reaction Wheel Calibration Response -0.25 N-m Commanded Torque

### 4.3 Vehicle MOI Test Results



(a) Reaction Wheels Rates



(b) Body Rates

Figure 4.2: Z-axis MOI Test Run

This section presents the results of the vehicle MOI tests described in Section 3.7.3. The results of the vehicle MOI testing were mixed because of an inability to eliminate

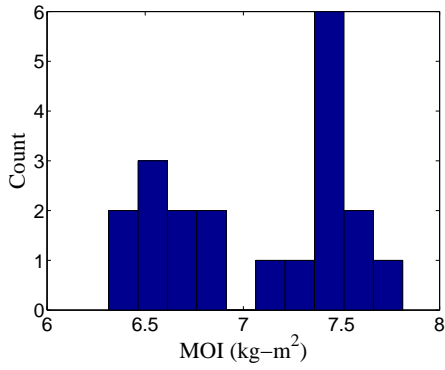
gravitational disturbance torques and the incorrect assumption that the  $X$ -,  $Y$ -, and  $Z$ -axes are the principal axes. Figure 4.2 shows the measurements captured during a typical test run performed about the  $Z$ -axis. It is important to note movement about both the  $X$ - and  $Y$ -axes, which indicated the presence of disturbance torques, products of inertia, misalignment of the reaction wheel, or all three. The time window of 2.5 sec to 4.5 sec was used to calculate the angular acceleration of both the reaction wheel and SimSat using MATLAB’s ‘polyfit’ algorithm and specifying a linear (first-order) fit. The mean and standard deviation were calculated for the 20 test runs in each direction, and histograms of the data was plotted to locate trends and outliers. Figures 4.3 through 4.5 show the the results of the testing sets, with the final mean and standard deviation provided in Table 4.1. The difference between the positive and negative torque tests indicate that despite balancing efforts, a significant gravitational disturbance torque was present in nearly all of the tests. Additionally, movement about all three axes was observed in most tests, indicating that the defined body axes  $X$ -,  $Y$ -, and  $Z$ -axes are close to, but not actually the principal axes and there are non-zero products of inertia for the chosen body axis set. The presence of non-zero products of inertia invalidates the assumption that  $X$ -,  $Y$ -, and  $Z$ -axes are principal axes, but the effects due to the products inertia could not be decoupled from disturbance torques and were deemed to be negligible for the remainder of the research effort. Accurately measuring the full MOI matrix including products of inertia, and methods to minimize the gravitational disturbance torques are suggested as future research efforts in Section 5.2.

Table 4.1: MOI Test Results

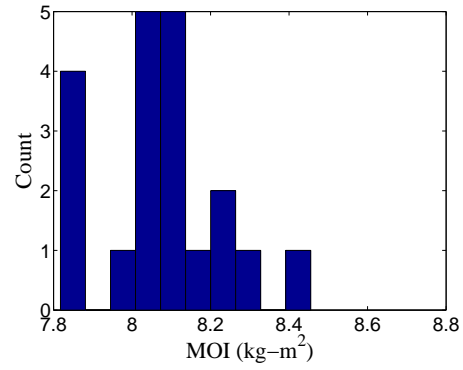
Axis	Mean (kg-m <sup>2</sup> )	St Dev
$X$	7.58	0.615
$Y$	8.12	0.256
$Z$	13.15	0.166

#### 4.4 CMG Gimbal Test Results

This section presents the results of the CMG gimbal testing described in Section 3.7.4. The CMG gimbal tests validated the performance of the gimbal motors by

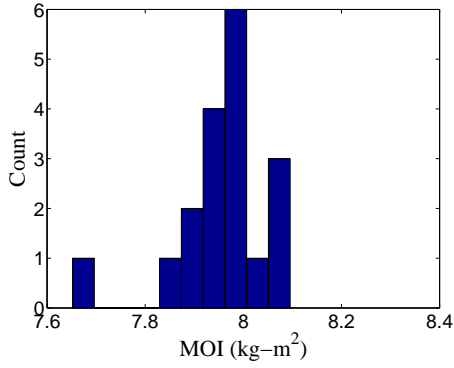


(a) X-axis +0.1 N-m Nominal Torque

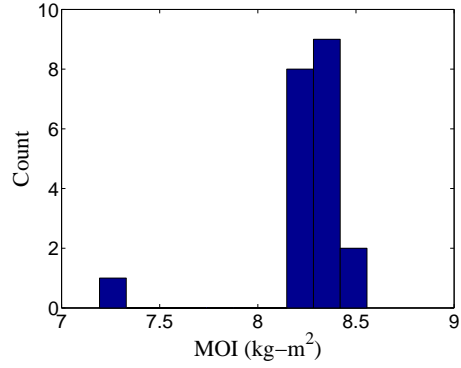


(b) X-axis -0.1 N-m Nominal Torque

Figure 4.3: X-axis MOI Tests Results

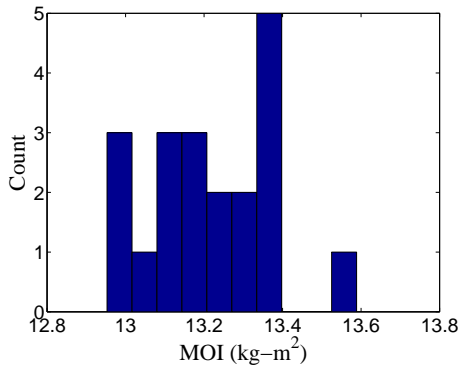


(a) Y-axis +0.1 N-m Nominal Torque

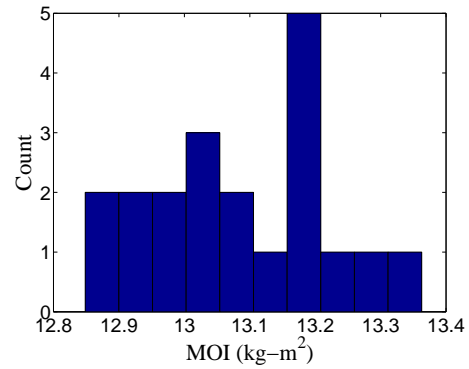


(b) Y-axis -0.1 N-m Nominal Torque

Figure 4.4: Y-axis MOI Tests Results



(a) Z-axis +0.1 N-m Nominal Torque



(b) Z-axis -0.1 N-m Nominal Torque

Figure 4.5: Z-axis MOI Tests Results

comparing the shaft encoder measurements with the rates commanded to the gimbal motor EPOS controller. The shaft encoder rate measurements were differentiated in real-time by the MicroAutoBox using the algorithm described in Section 3.6.1.2. The shaft encoder measurements were also differentiated in post-processing using MATLAB’s linear fit algorithm. Figure 4.6 shows the test results for gimbal 1 at  $\pm 1$  rad/sec. As expected, the real-time differential is noisy, but bounded and the results clearly indicate that the commanded and measured values are equal, therefore the gimbal motors are functioning as intended. Thus, all gimbal angle measurements were taken from the shaft encoders, while gimbal rates were measured using the gimbal motor rate from the EPOS controller. Results from the tests performed on gimbals 2, 3, and 4 are in Appendix 1.3.

#### ***4.5 CMG and Reaction Wheel Rest-to-Rest Test Results***

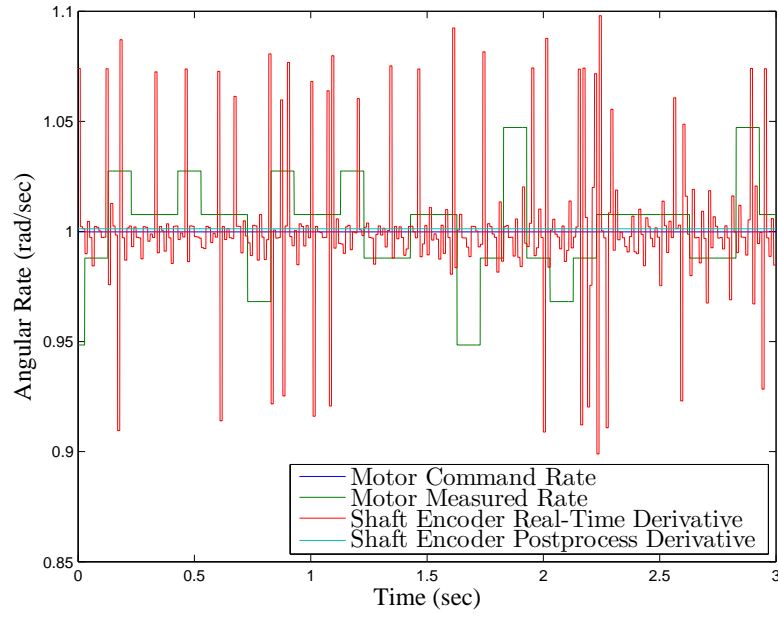
The position hold and rest-to-rest maneuver tests were conducted to validate SimSat’s ACS performance against the following performance criteria from Chapter I:

1. Positioning accuracy shall be  $\pm 0.01^\circ$ ,
2. A  $\pm 10^\circ$  rest-to-rest maneuver about the *X*- and *Y*-axis shall be demonstrated within 10 seconds,
3. A  $\pm 30^\circ$  rest-to-rest maneuver about the *Z*-axis shall be demonstrated within 10 seconds,
4. and SimSat’s angular velocity shall not exceed  $180^\circ/\text{sec}$ ,

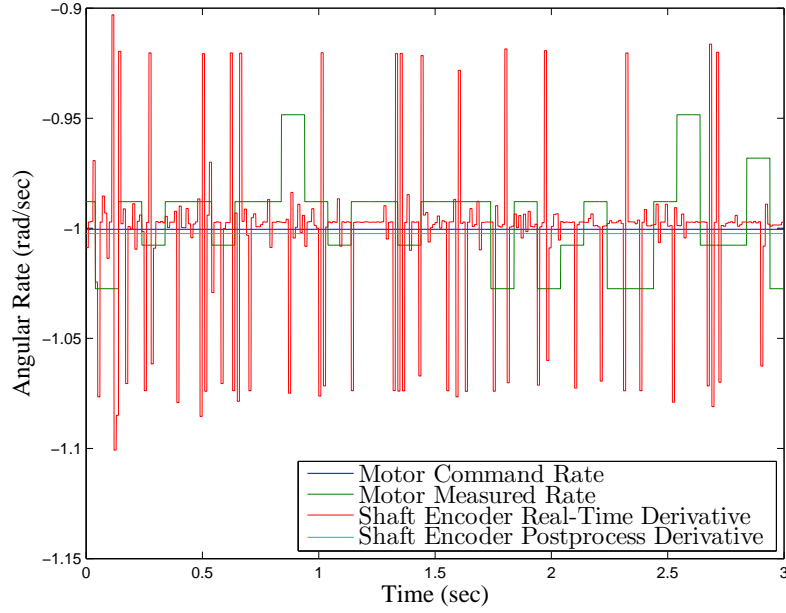
using the tests described in Section 3.7.6.

Before discussing the actual results, it is useful to compute the theoretical performance of SimSat for the slew maneuvers to provide a reference for comparison and ensure that SimSat can complete the maneuver within the required time. Based on the controller’s maximum torque command of  $\pm 0.25$  N-m, the theoretical minimum time slew can be computed by assuming ‘bang-bang’ torque application along a principal axis with no disturbance torques, resulting in

$$t_{slew} = \sqrt{\theta_{slew} \frac{I}{\tau}} \quad (4.1)$$



(a) 1 rad/sec clockwise



(b) 1 rad/sec counterclockwise

Figure 4.6: Gimbal 1 Verification



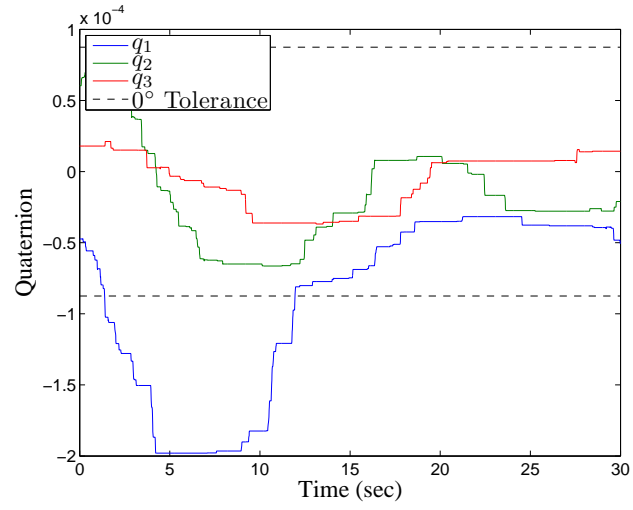
where  $t_{slew}$  is the time required to complete the slew,  $\theta_{slew}$  is the angle traveled in radians,  $\tau$  is the applied torque, and  $I$  is the moment of inertia about the axis of rotation. For a  $10^\circ$  slew, the minimum time required is 2.3 seconds; for a  $30^\circ$  slew the minimum time is 5.25 seconds.

The position hold test was performed using the reaction wheels and the CMG array with the Moore-Penrose Pseudoinverse Steering Law (MPPSL) with the gimbal initial angles set at  $\bar{\delta}_0=[0,0,0,0]$ . The rest-to-rest tests were performed on each axis in both directions using the reaction wheels and both the MPPSL and Generalized Inverse Steering Law (GISL) with different initial gimbal angles, as listed in Table 4.2, for a total of 30 reorientation tests, and 32 tests total. The  $X$ - and  $Y$ -axis tests produced similar results due to similar MOI and actuator configuration, therefore the  $Y$ -axis test results are provided in Appendix A1.4. The positive and negative direction tests also produced similar results so the negative slew tests are also in Appendix A1.4. The reaction wheel tests will be covered first, followed by the CMG position hold,  $X$ -axis, and finally  $Z$ -axis tests. Additionally, only the quaternion vector  $\vec{q}$ , is plotted for angular orientation as  $q_4$  remains close to 1 in all tests and does not provide additional insight into the response behavior.

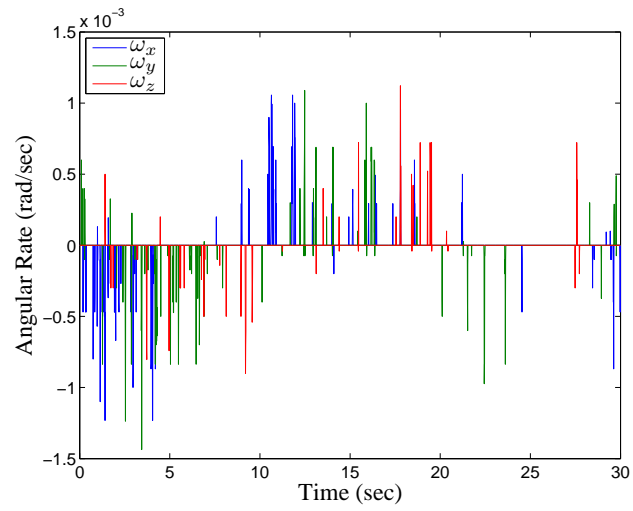
Table 4.2: Rest-to-Rest Actuator Configurations

Actuator	Steering Law	Initial Gimbal Angles $\bar{\delta}_0$
Reaction Wheels	N/A	N/A
CMG	MPPSL	$[0,0,0,0]$
CMG	MPPSL	$[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$
CMG	GISL	$[0,0,0,0]$
CMG	GISL	$[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

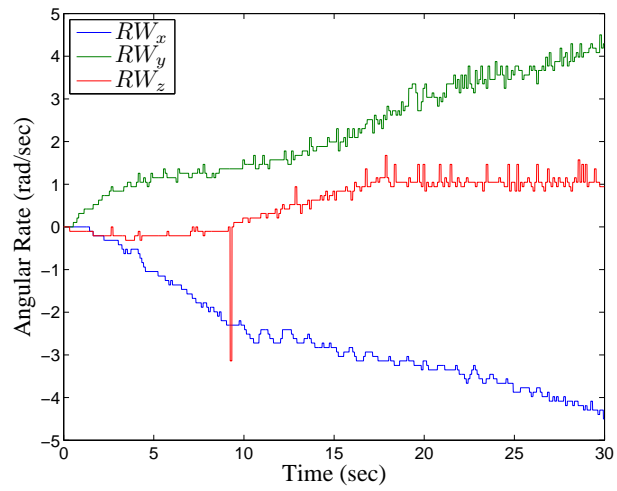
*4.5.1 Reaction Wheel Rest-to-Rest Test Results.* In order to meet the position hold requirement of  $\pm 0.01^\circ$ , the ACS must hold  $q_1$  through  $q_3$  to less than  $8.75 \times 10^{-5}$  for the duration of the test, as indicated on by the bounding bars in Fig. 4.7(a). The  $X$ -axis error,  $q_1$  clearly has an initial transient that requires the ACS approximately 12 sec to address, but the system is able to drive this error to within the tolerance. The initial transient is likely the result of multiple error sources, including gyroscope noise shown



(a) Quaternion Vector



(b) Body Rates



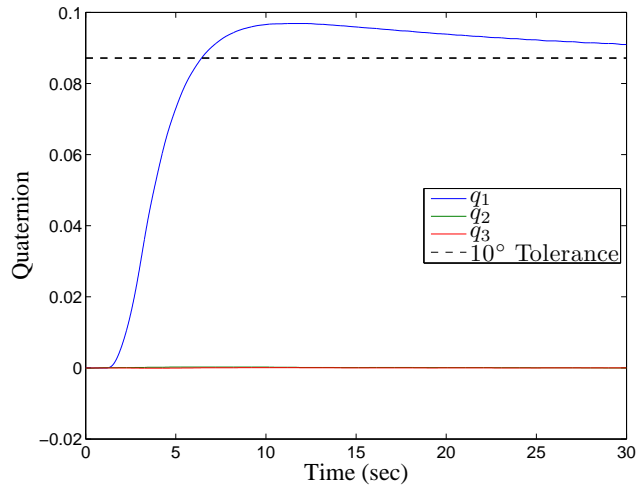
(c) Reaction Wheel Rates

Figure 4.7: Reaction Wheel Position Hold

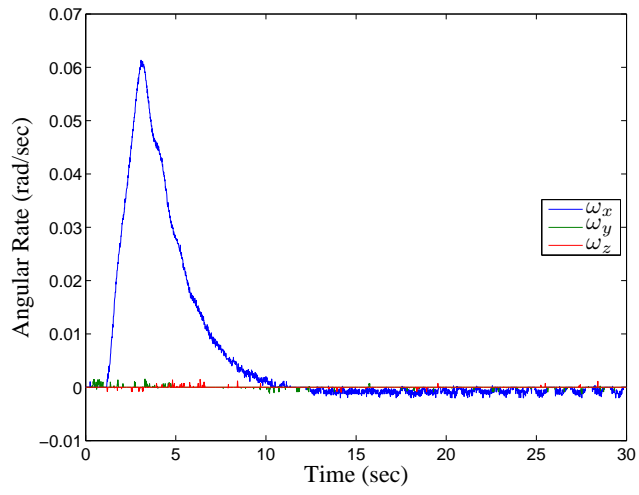
in Fig. 4.7(b), misalignments of the reaction wheels and LN-200, and phase lag of the reaction wheels. There is also a minor signal anomaly on the  $Z$ -axis wheel at  $\tilde{9}$  seconds which is likely the result of noise on the CAN bus. Additionally, Fig. 4.7(c) indicates a small, but measurable accumulation of angular momentum about the both the  $X$ - and  $Y$ -axes, indicating disturbance torque.

The  $X$ -axis slew test required a  $+10^\circ$  slew, requiring that  $q_1$  reach a value of  $+8.72 \times 10^{-2} \pm 8.7 \times 10^{-5}$  while  $q_1$  and  $q_3$  remain zero. Figure 4.8 shows the results of this test. SimSat fails to reach the desired orientation within the 30 second sampling window but clearly overshoots the target as indicated in Fig. 4.8(a). The rapid rise followed by slow decay decay seen in Fig. 4.8(a) is indicative of integrator windup caused by restricting the reaction wheel torque to 0.25 N-m as discussed in Section 2.5.1. Adjustments to the ACS tuning should allow SimSat to reach the target within the 10 second requirement. Additionally, the rise time is nearly double the theoretical time for SimSat to complete the maneuver, indicating either disturbance torques or that the MOI estimates are incorrect. Fig. 4.8(c) shows the  $X$ -axis reaction wheel accumulates momentum throughout the test indicating a disturbance torque about the  $X$ -axis. Finally, the  $X$ -axis reaction wheel rates and body rates display an oscillation which grows in magnitude starting about 17 seconds into the test. The oscillation indicates that the control system with these gains may not be stable and further evaluation is suggested.

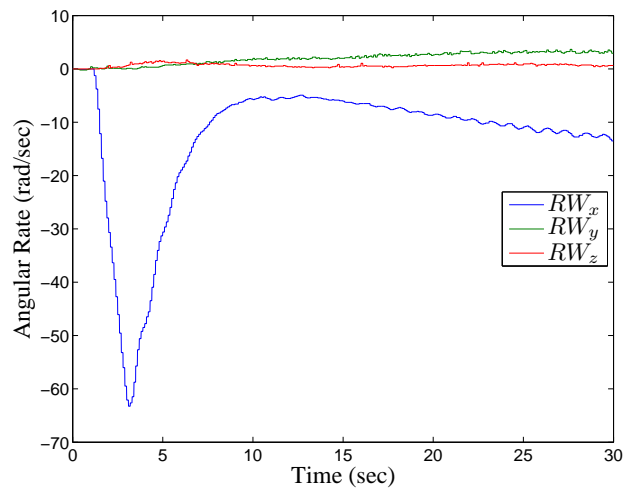
The  $Z$ -axis slew test involved a  $+30^\circ$  slew, requiring that  $q_3$  reach a value of  $+2.61 \times 10^{-1} \pm 8.7 \times 10^{-5}$  while  $q_1$  and  $q_2$  remain zero. Figure 4.9 shows the results of this test. Like the  $X$ -axis slew, SimSat overshoots the target at 9 seconds into the test but fails to settle in the allotted time, shown in Fig. 4.9(a). The vehicle does however end very close to the target, indicating that the gains for the  $Z$ -axis are better tuned. Additionally, there is a clear  $Y$ -axis transient at the start of the test, shown in Fig 4.9(b) that SimSat must overcome, and a second transient along the  $X$ -axis that occurs when the  $Z$ -axis reaction wheel first starts. The cause of this transient is unknown, but is likely due to either misalignment of the reaction wheel causing coupling between the  $X$ - and  $Z$ -reaction wheels cross-coupling due to the  $Z$ -axis not being one of the principal axes. Lastly, as in previous tests, the  $X$ - and  $Y$ -reaction wheels end the test with stored



(a) Quaternion Vector

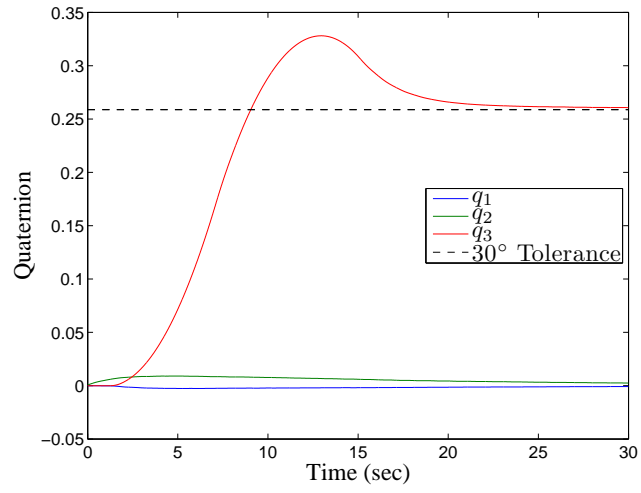


(b) Body Rates

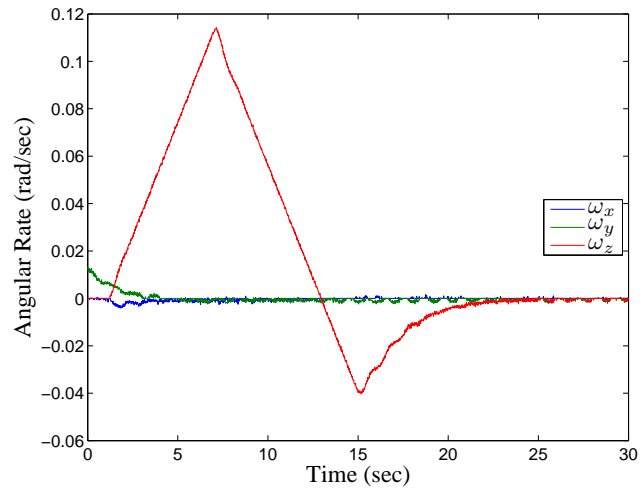


(c) Reaction Wheel Rates

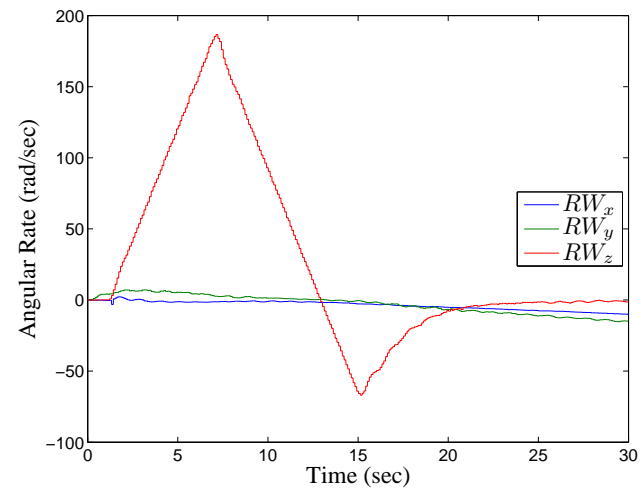
Figure 4.8: Reaction Wheel  $10^\circ$  X-axis Slew



(a) Quaternion Vector



(b) Body Rates



(c) Reaction Wheel Rates

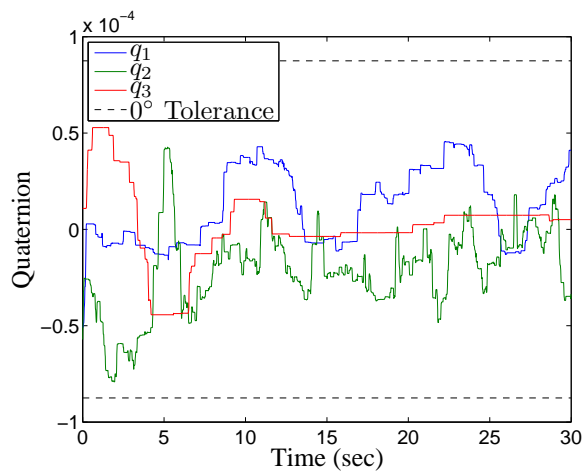
Figure 4.9: Reaction Wheel  $30^\circ$  Z-axis Slew

angular momentum, indicating the presence of disturbance torques. Finally, there is an oscillation in the  $Y$ -axis reaction wheel measurements and  $Y$ -axis body rates. A similar oscillation was encountered during the tuning phase which was attributed to the phase lag in the reaction wheel system.

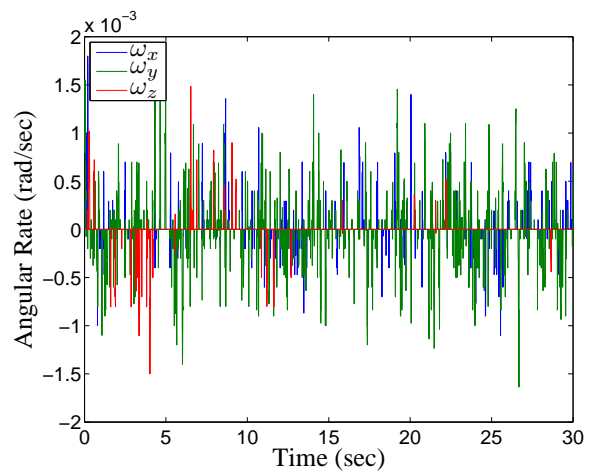
Summarizing the reaction wheel rest-to-rest maneuver tests, it is clear that the reaction wheels are capable of providing precision maneuvering, however the PID control should be adjusted if faster settling times are required. The presence of oscillations in the  $Z$ -axis slew test indicate that the gain settings used may have stability issues due to the interaction between the controller and the reaction wheel actuators, and SimSat's body dynamics. The PID control system needs to be re-tuned to provide stability and compensate for the reaction wheel actuator phase lag.

*4.5.2 CMG Rest-to-Rest Test Results.* The CMG position hold test was performed using the MPPSL, requiring the vehicle maintain its attitude to within  $\pm 0.01^\circ$ ; in quaternions maintain  $q_1$  through  $q_3$  to less than  $8.75 \times 10^{-5}$  for the duration of the test. The results, shown in Fig. 4.10(a), indicate that the CMG array ACS was able to hold SimSat at this orientation for the 30 second test. Figures 4.10(c) and (d) show that the gimbals moved slowly but steadily to counteract a small disturbance torque, primarily about the  $Y$ -axis based on the angular momentum stored by the CMG array at the end of the experiment. Because the gimbals did not move a significant amount during this test no singularities were encountered, and therefore the GISL and alternate initial gimbal initial conditions  $\bar{\delta}_0$  were not tested.

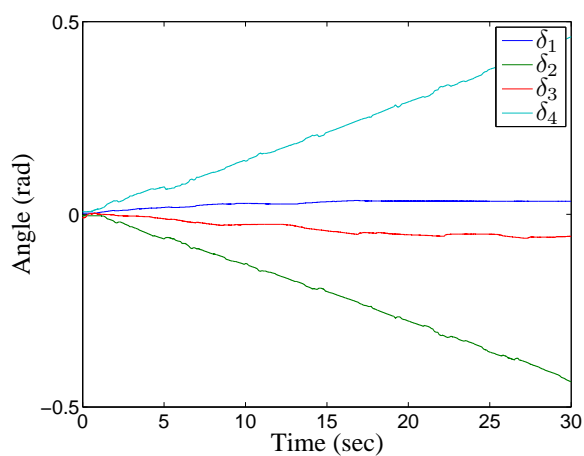
*4.5.2.1 CMG X-axis Slew Maneuvers.* The  $X$ -axis maneuver requirement is to slew  $10^\circ$  and settle to within  $\pm 0.01^\circ$  in 10 seconds or less, with the quaternion equivalent of having  $q_1$  reach a value of  $+8.72 \times 10^{-2} \pm 8.7 \times 10^{-5}$  while  $q_2$  and  $q_3$  remain zero. The results of the test conducted with  $\bar{\delta}_0 = [0, 0, 0, 0]$ , shown in Fig. 4.11(a) indicate that the ACS reached the desired slew angle, overshoot, and failed to settle in time. As with the reaction wheels, the CMG ACS displays signs of integrator windup adversely affecting performance. Figure 4.11(c) and (d) indicate that the CMG array may have encountered a singularity shortly before 5 seconds into the test, but the array was able to pass



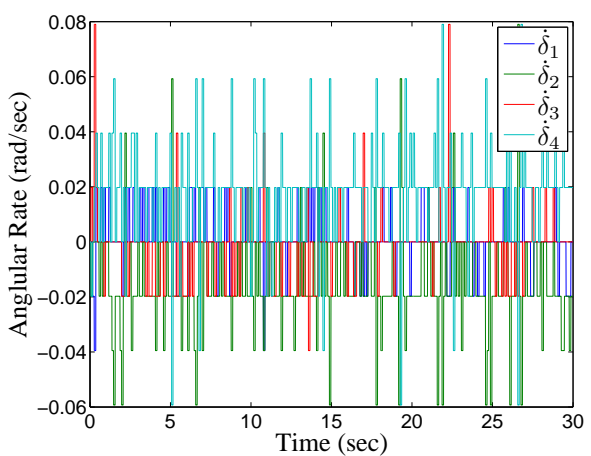
(a) Quaternion Vector



(b) Body Rates



(c) Gimbal Angles



(d) Gimbal Rates

Figure 4.10: CMG Position Hold, MPPSL,  $\bar{\delta}_0 = [0, 0, 0, 0]$

through it without affecting the orientation. Note that the gimbal angle measurements wrap from  $-\pi$  to  $\pi$  but the gimbal has unrestricted motion. Figure 4.11(c) also shows that instead of returning to their original configuration, gimbals 1 and 3 completed the slew maneuver by transitioning  $\pi$  rads, or  $180^\circ$  away from their starting configuration. Additionally, despite being a rest-to-rest maneuver, the CMG array did accumulate a stored angular momentum of  $[-0.0552, -0.3941, 0.0546]$  N-m-s or a magnitude of 0.4017 N-m-s, indicating disturbance torques acting on the vehicle.

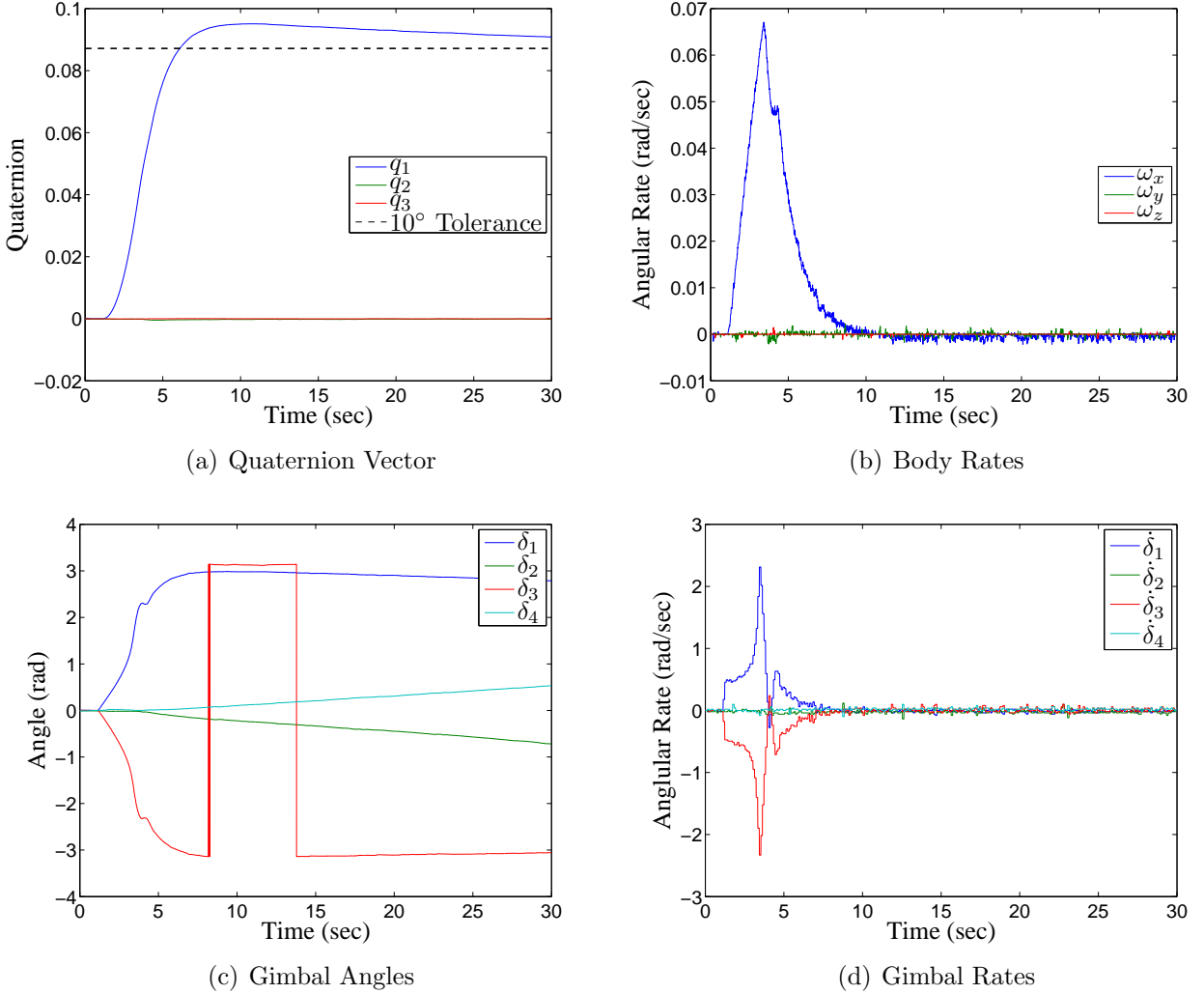


Figure 4.11: CMG  $10^\circ$  X-axis Slew, MPPSL,  $\bar{\delta}_0=[0,0,0,0]$

The  $10^\circ$  X-axis slew was repeated using the MPPSL with the  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , and the results shown in Fig. 4.12. The attitude and body rate responses shown in Fig. 4.12(a) and (b) are nearly identical to those seen in Fig. 4.11(a) and (b), except



for a small transient at about 1 second after the movement is commanded. Unlike the previous test, starting with  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$  leads to all CMGs being involved in the movement, rather than only two as observed in the previous test with  $\bar{\delta}_0 = [0,0,0,0]$ .

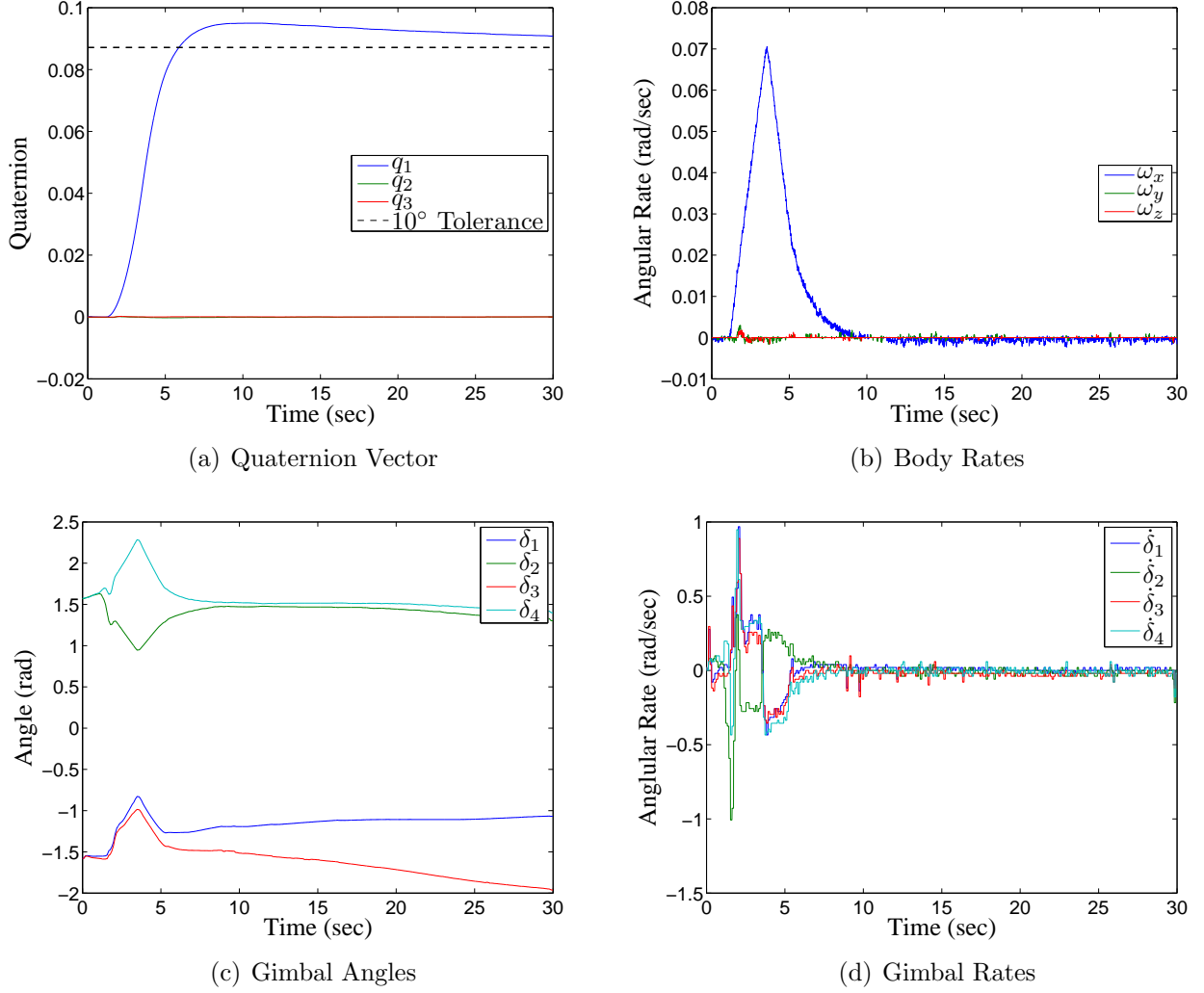


Figure 4.12: CMG 10° X-axis Slew, MPPSL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

The 10° X-axis slew was repeated using the GISL using  $\bar{\delta}_0 = [0,0,0,0]$  and  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$  with results shown in Figs. 4.13 and 4.14, respectively. Figure 4.13 shows that the GISL tried to couple null motion to move gimbals 2 and 4 as gimbals 1 and 3 approach  $\frac{\pi}{2}$ , leading to a temporary shift in the vehicle's orientation seen shortly before 5 seconds into the test. Comparing the result with the MPPSL maneuver for the same conditions, the MPPSL appears to have smoother control and fewer deviations from the shortest path. Comparing the GISL and MPPSL solutions for  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ ,

the GISL test appears to encounter a singularity near the end of the test, indicated in Fig. 4.14(d) by the rapid oscillations in the gimbal rates that begin approximately 27 seconds into the test.

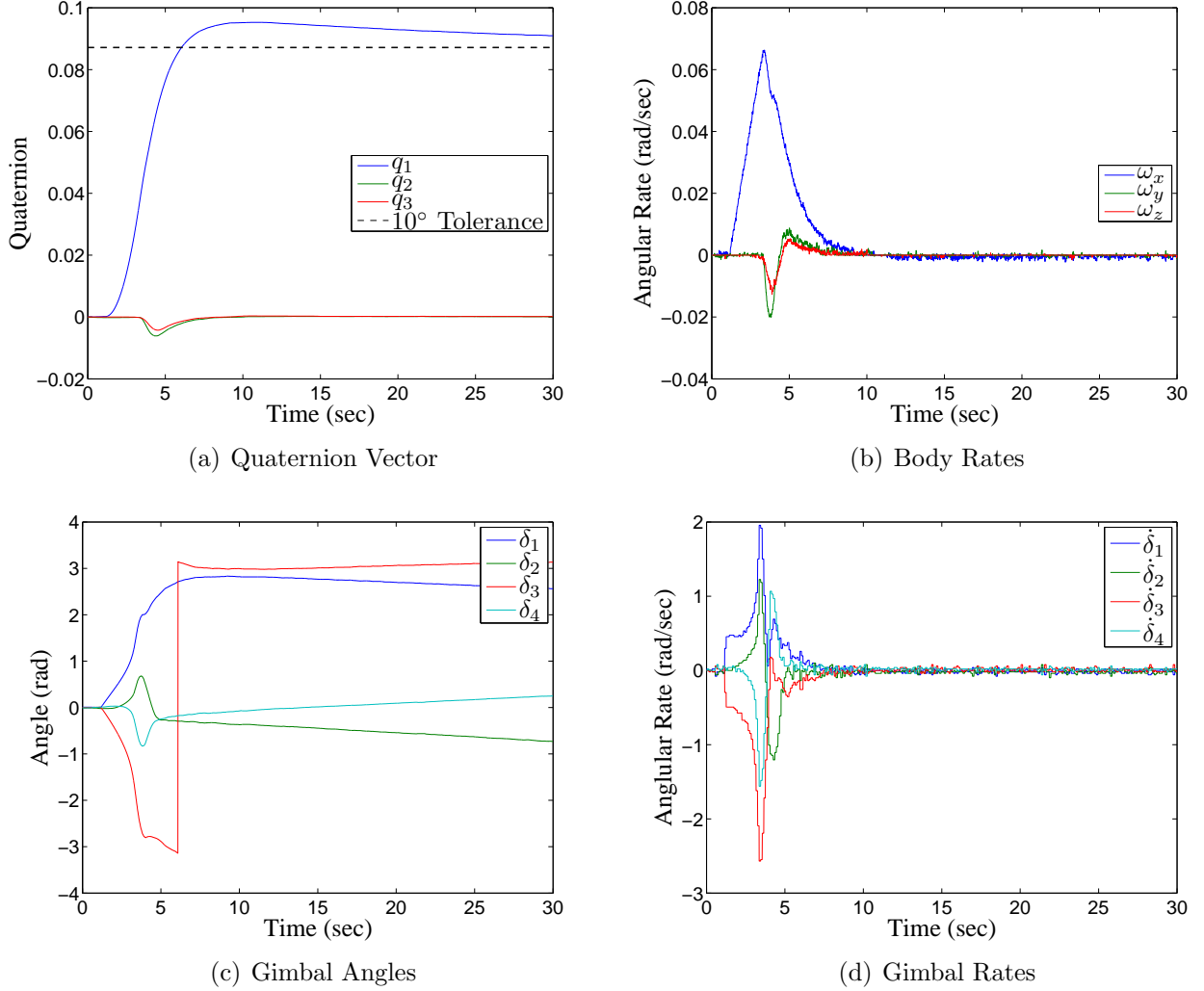
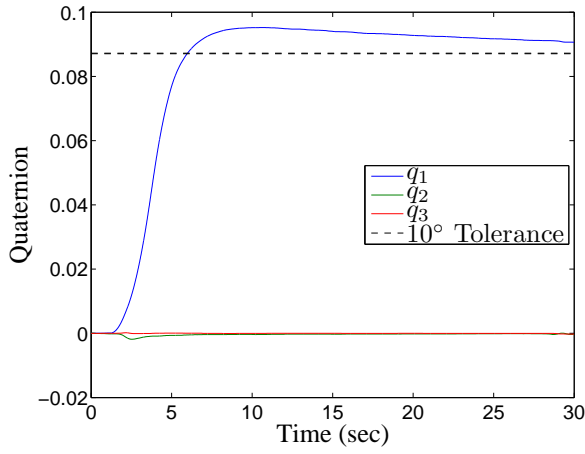
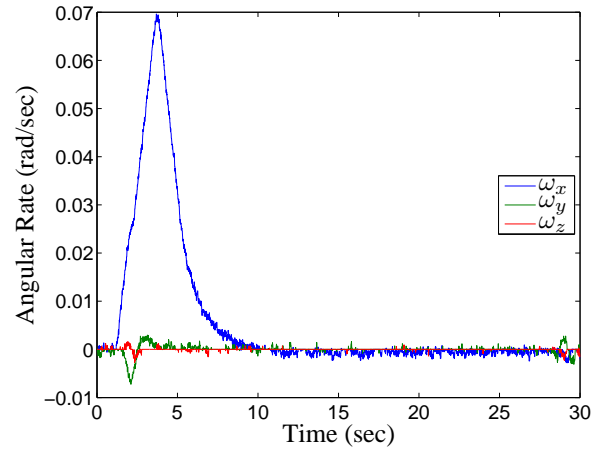


Figure 4.13: CMG 10° X-axis Slew, GISL,  $\bar{\delta}_0=[0,0,0,0]$

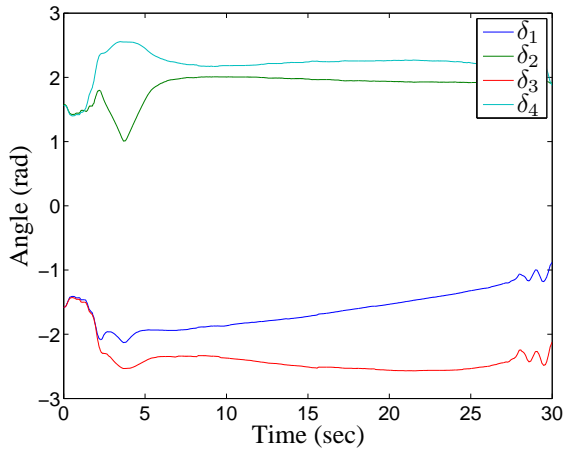
**4.5.2.2 CMG Z-axis Slew Maneuvers.** The Z-axis slew test required SimSat to slew 30° about the Z-axis to have  $q_3$  equal  $+2.61 \times 10^{-1} \pm 8.7 \times 10^{-5}$  and maintain  $q_1$  and  $q_2$  equal to zero. Figure 4.15 shows the results using MPPSL with  $\bar{\delta}_0=[0,0,0,0]$ , with Fig. 4.15(a) showing that SimSat nearly reached the target at 20 seconds into the test. Analytical results show that the  $\pm 0.01^\circ$  requirement was not satisfied, however it is clear that the control gains are fairly good for this maneuver range. Analyzing



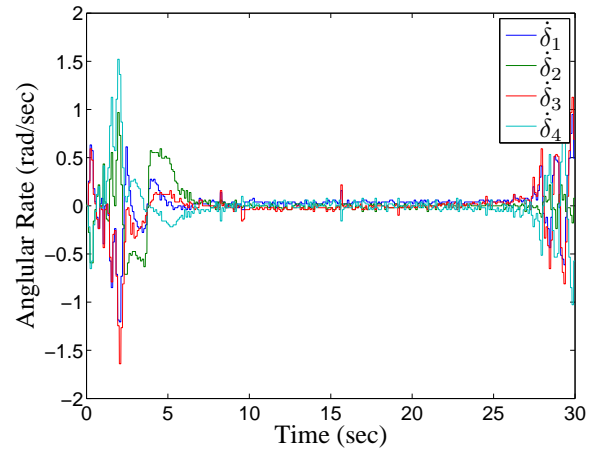
(a) Quaternion Vector



(b) Body Rates



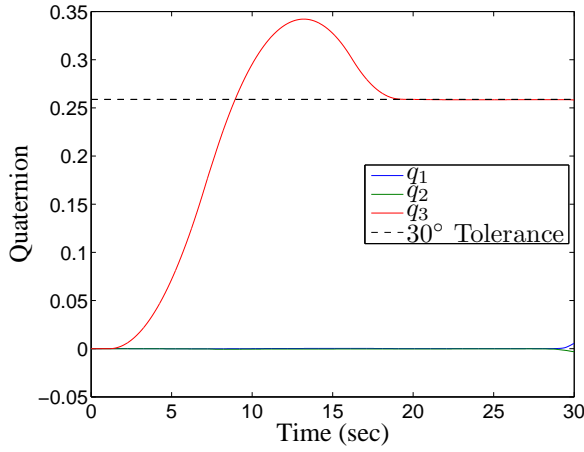
(c) Gimbal Angles



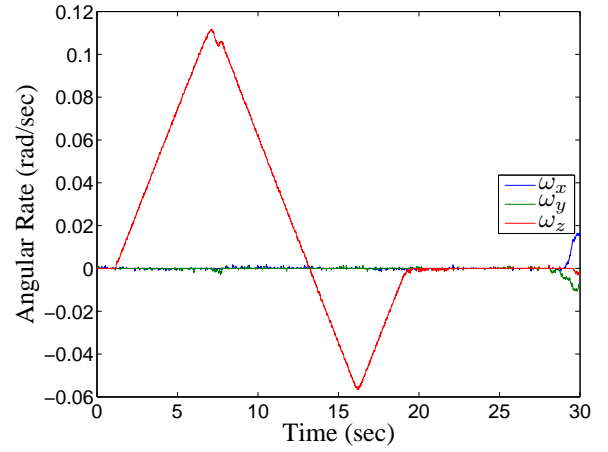
(d) Gimbal Rates

Figure 4.14: CMG  $10^\circ$  X-axis Slew, GISL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

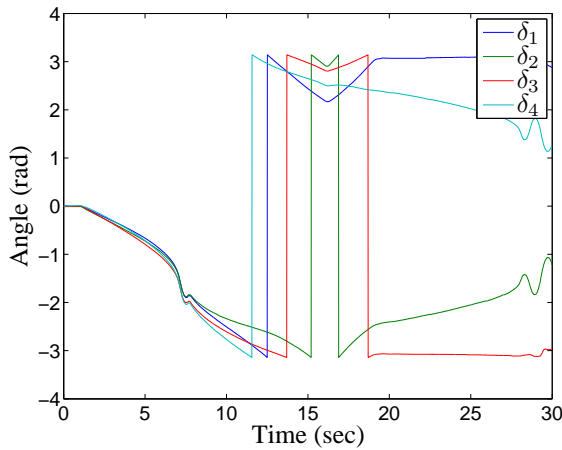
Fig. 4.15(c), the algorithm reached saturation of  $\bar{\delta}=[-\frac{\pi}{2},-\frac{\pi}{2},-\frac{\pi}{2},-\frac{\pi}{2}]$  about 7 seconds into the test, maintained saturation briefly, then continued to move the gimbal angles to  $\bar{\delta}=[\pi,\pi,\pi,\pi]$  rather than reversing direction and returning to the initial condition of zero. It is important to note that the gimbal angle measurement wraps around from  $-\pi$  to  $\pi$  and mathematically the angles are equal. As with previous tests, there is an accumulation of angular momentum in the system, in this test  $[-0.1286, -0.4998, 0.0369]$  N-m-s or a magnitude of 0.5174 N-m-s, indicating the presence of disturbance torques and there are indications that these disturbances pushed the system toward a singularity, as indicated by the oscillating gimbal rates in Fig. 4.15(d) around 27 seconds into the test.



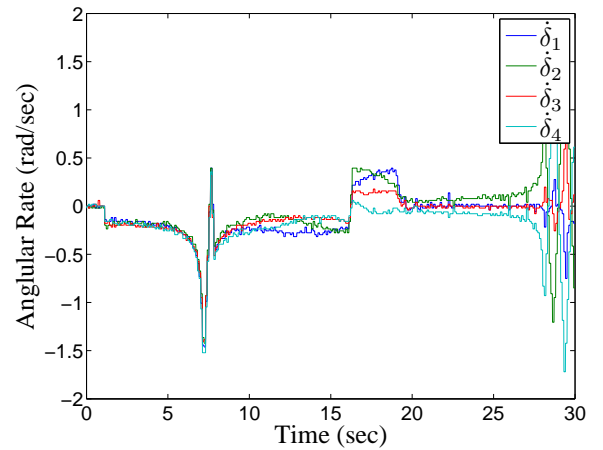
(a) Quaternion Vector



(b) Body Rates



(c) Gimbal Angles



(d) Gimbal Rates

Figure 4.15: CMG 30° Z-axis Slew, MPPSL,  $\bar{\delta}_0=[0,0,0,0]$

Applying the same  $Z$ -axis slew using the MPPSL with  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , shown in Fig. 4.16, resulted in vehicle movements similar to those in Fig. 4.15(a) and (b), but drastically different gimbal movements. At approximately 5 seconds into the test, the gimbals approach a singular state as indicated by the large gimbal rates in Fig. 4.16(d), but passed through the singularity before approaching saturation. Comparing the two initial conditions,  $\bar{\delta}_0 = [0, 0, 0, 0]$  was able to saturate the array in less time and achieve a faster overall response because it did not have to pass through a singularity.

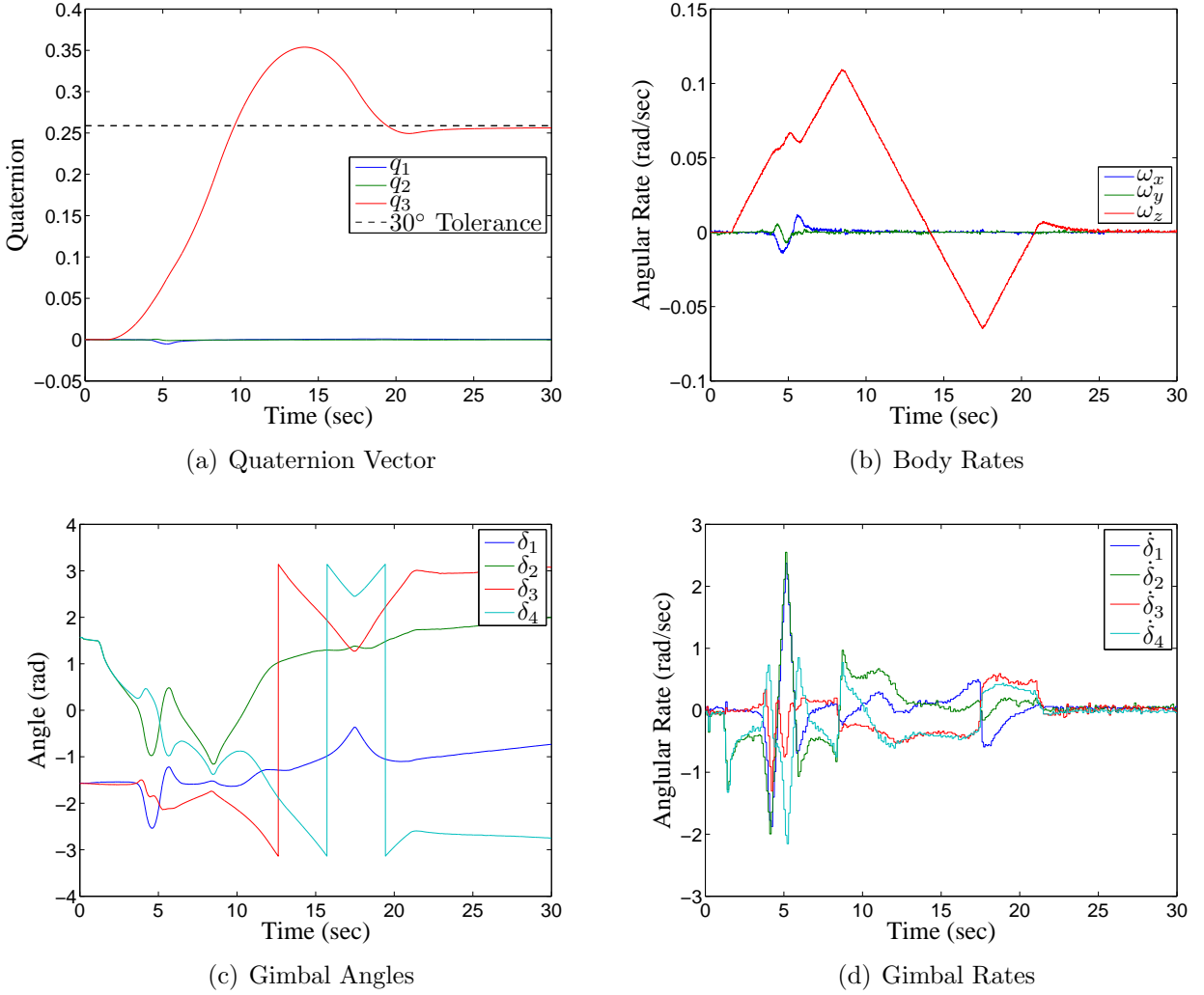
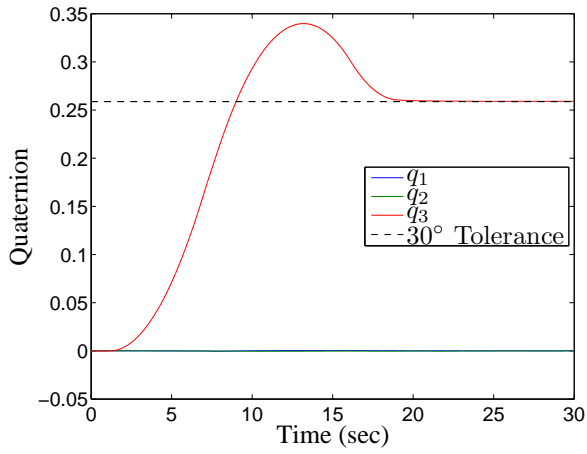
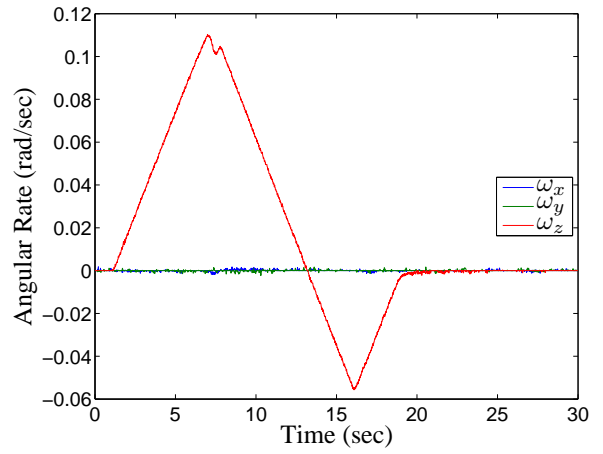


Figure 4.16: CMG 30° Z-axis Slew, MPPSL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

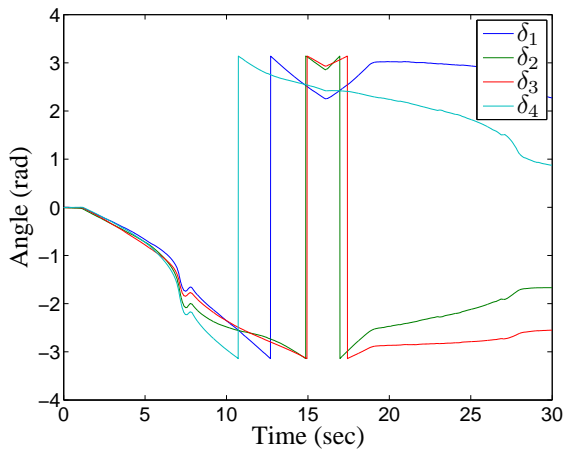
Figures 4.17 and 4.18 show the results from the  $Z$ -axis slew test using the GISL with the two initial conditions. Comparing Fig. 4.17 with 4.15, the responses are nearly identical which is expected because there are no singularities on the path between the



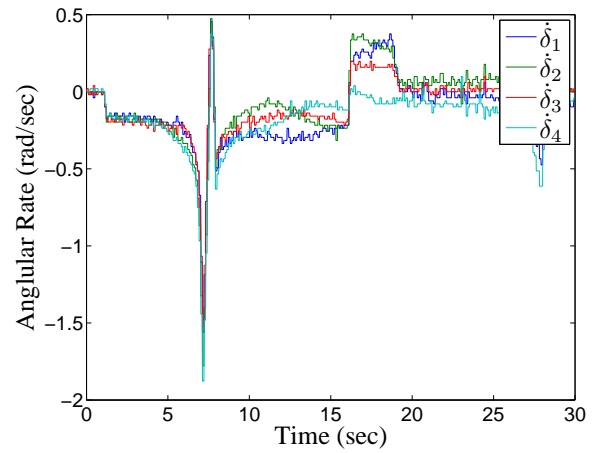
(a) Quaternion Vector



(b) Body Rates

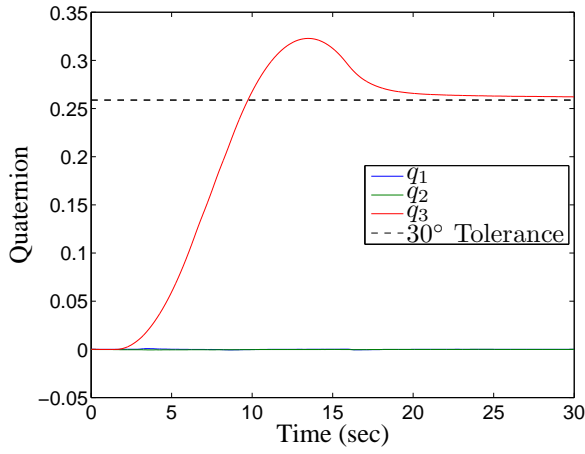


(c) Gimbal Angles

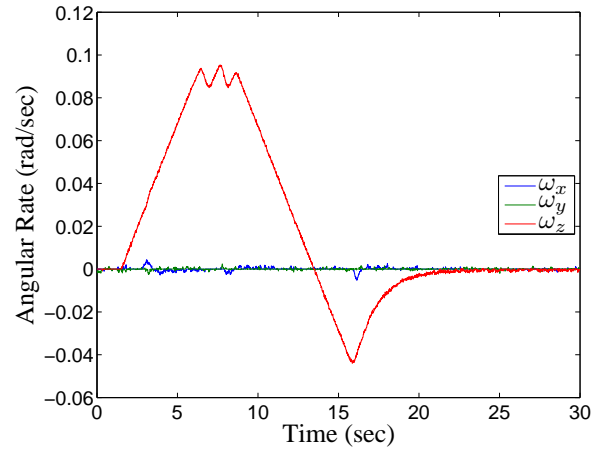


(d) Gimbal Rates

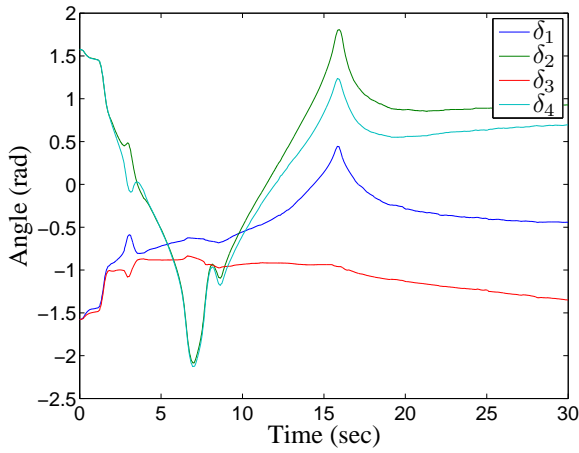
Figure 4.17: CMG  $30^\circ$  Z-axis Slew, GISL,  $\bar{\delta}_0 = [0, 0, 0, 0]$



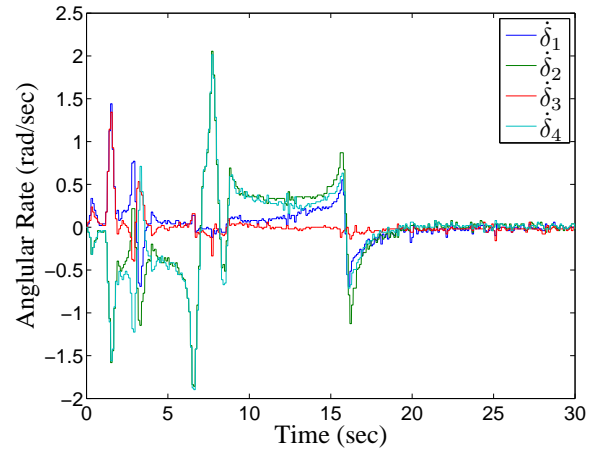
(a) Quaternion Vector



(b) Body Rates



(c) Gimbal Angles



(d) Gimbal Rates

Figure 4.18: CMG 30° Z-axis Slew, GISL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

initial condition of  $\bar{\delta}_0=[0,0,0,0]$  and a  $Z$ -axis saturation of  $\bar{\delta}=[-\frac{\pi}{2},-\frac{\pi}{2},-\frac{\pi}{2},-\frac{\pi}{2}]$ . The GISL does not perform as well with the starting condition of  $\bar{\delta}_0=[\frac{\pi}{2},-\frac{\pi}{2},\frac{\pi}{2},-\frac{\pi}{2}]$ , however it does not enter the singularity that the MPPSL enters when starting from the same initial conditions. Figure 4.17(c) also shows that the GISL never approached the  $Z$ -axis saturation point, as gimbals 1 and 3 move away from  $\frac{\pi}{2}$  and do not return.

*4.5.2.3 CMG and Reaction Wheel Slew Test Conclusions.* Overall, the slew maneuver tests indicated that both the reaction wheel ACS and the CMG ACS are capable of providing precision maneuvering in the face of small disturbance torques. In all tests, disturbance torques adversely affected the overall performance. Because the reaction wheels are uncoupled and independent of one another the disturbance torques only impacts the performance of the axis which the torque is applied. For example an  $X$ -axis disturbance has no significant impact on the  $Z$ -axis reaction wheel performance. The CMGs, however, must work together as an array to generate the desired maneuvering torque and counteract the disturbances. Because the CMG array must function as a single unit, any disturbance torque prevents the CMG array from reaching the ideal saturation state for a given maneuver because the CMGs array must also store sufficient angular momentum to offset the disturbance torque. Additionally, the CMG ACS appears to have better stability and a faster overall response, provided the CMG array does not encounter a singularity. The reaction wheel ACS, by comparison, is slow and shows indications of instability with the control settings tested, but does not suffer from singularities.

## **4.6 CMG and Reaction Wheel Torque Test Results**

CMG and reaction wheel torque testing was performed using the actuator configurations as the rest-to-rest maneuvers, listed in Table 4.2, with three different masses placed 0.6 m from SimSat's center of mass, based on placing the mass at 14.28 cm from the edge of SimSat's main deck plate, along the negative  $X$ -,  $Y$ -, and  $Z$ -axes:

1. 17.2 g - 0.101 N-m
2. 33.8 g - 0.199 N-m



### 3. 42.6 g - 0.250 N-m

Additionally, a no load control test was conducted. All three reaction wheels displayed similar behavior for all three masses, therefore only the  $X$ -axis results for the control test, 17.2 g, and 42.6 g are presented in Section 4.6.1. The complete set of test results are available in Appendix A1.5. The CMGs displayed similar behavior about both the  $X$ - and  $Y$ -axes for all three masses, but different behavior for the two steering laws and initial gimbal angles. Additionally, the CMGs had different behavior about the  $Z$ -axis. The results for the  $X$ - and  $Z$ -axes tests are presented for the 17.2 g and 42.6 g tests for the four steering law and initial gimbal angle combinations are presented in Section 4.6.2 with the remaining test results available in Appendix A1.5.

*4.6.1 Reaction Wheel Torque Test Results.* The results of the reaction wheel no load control test are shown in Fig. 4.19 and indicated that a small disturbance torque was acting on the vehicle in both the  $X$ - and  $Y$ -axes. This disturbance torque is most likely the gravitational disturbance torque caused by a small misalignment between the center of mass and the center of rotation. The net angular momentum accumulated due to the disturbance torques is  $5.84 \times 10^{-2}$  N-m-s, indicating an average disturbance torque of  $+1.20 \times 10^{-3}$  N-m about the  $X$ -axis,  $-1.619 \times 10^{-3}$  N-m about the  $Y$ -axis, and  $+3.73 \times 10^{-4}$  N-m about the  $Z$ -axis.

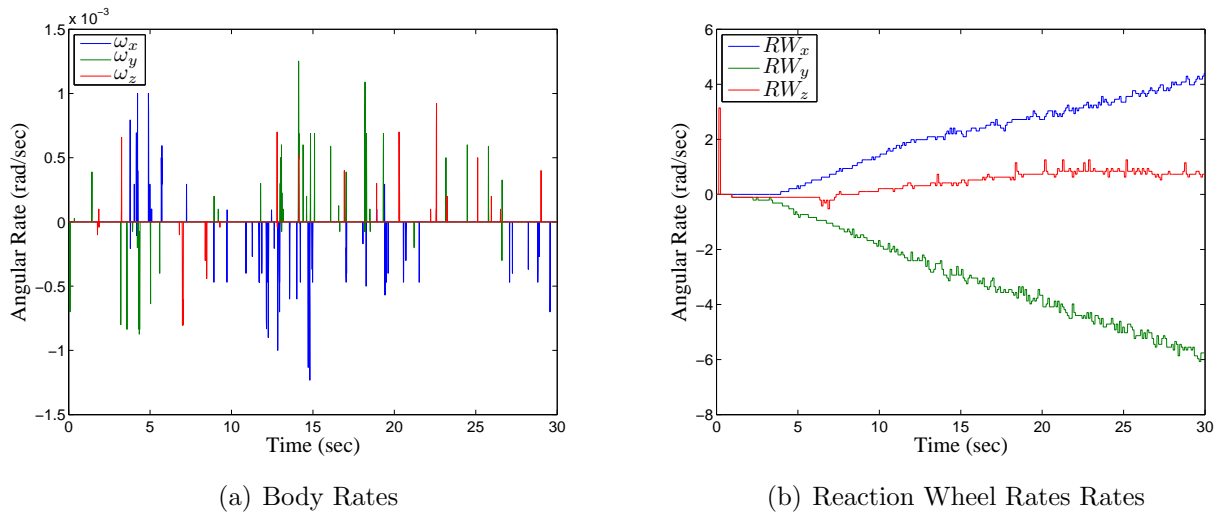
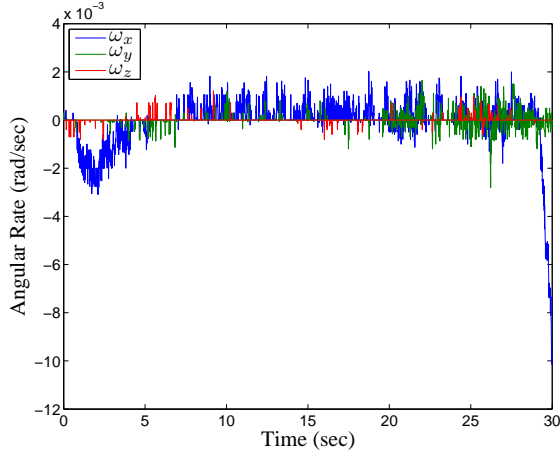


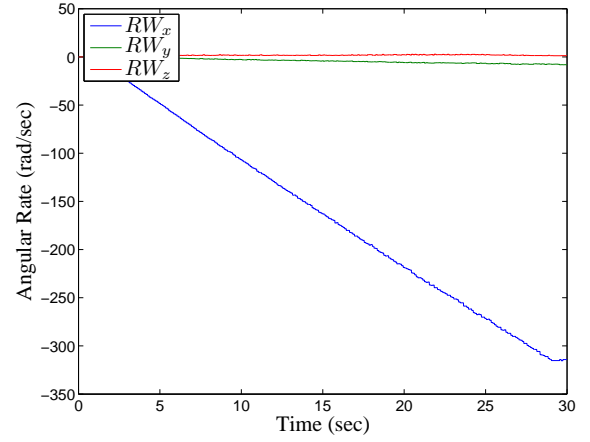
Figure 4.19: Reaction Wheel Disturbance Control Test with 0 g

The results of the  $X$ -axis torque test with 17.2 g for an estimated torque of -0.101 N-m are shown in Fig. 4.20. The initial rotation rate in the  $X$ -axis is caused by the change in controller settings from the fan/thruster, mode used to stabilize SimSat during the test setup, and the reaction wheel mode. The disturbance seen during the test is caused by the differing amount of integrated error required by the reaction wheels to provide the nominal 0.101 N-m of torque compared to the amount of integrated error required by the fan/thrusters. The initial disturbance is less than  $3 \times 10^{-3}$  rad/sec and is quickly damped out. The  $X$ -axis reaction wheel reaches saturation about 29 seconds into the test, indicated in Fig. 4.20(a) by SimSat acquiring a significant  $X$ -axis disturbance, and shown in Fig. 4.20(a) by the reaction wheel rate maintaining a constant 314 rad/sec and no longer applying torque. At saturation, the reaction wheel had accumulated 2.49 N-m-s of angular momentum. The torque applied by the reaction wheel calculated using the measured acceleration is -0.0883 N-m. Accounting for the disturbance torque the applied torque should have been -0.0998 N-m, indicating an 11% error in the torque generated by the reaction wheels compared to the torque applied. This difference could be the combined result of SimSat's center of mass shifting from that of the control test which would alter the gravitational disturbance torque, as well as a potential error in the location of the applied mass relative to the center of rotation due to imperfections in the size of SimSat's main deck plate.

Figure 4.21 shows the results of the  $X$ -axis test with 42.6 g for an estimated applied torque of -0.250 N-m. Accounting for disturbance torques, the applied torque was -0.249 N-m. The 42.6 g test was limited to 10 seconds to prevent saturation and ensure SimSat would not contact the pedestal and alter its center of mass due to the impact. SimSat acquired a small rotational transient as the control system switched from fan/thrusters to reaction wheels for stabilization. The torque applied by the reaction wheel based on its angular acceleration was -0.226 N-m or approximate error of 9% which is consistent with the previous test. The results for the  $X$ -axis test with 33.8 g, and all the  $Y$ -axis and  $Z$ -axis tests yielded similar results.

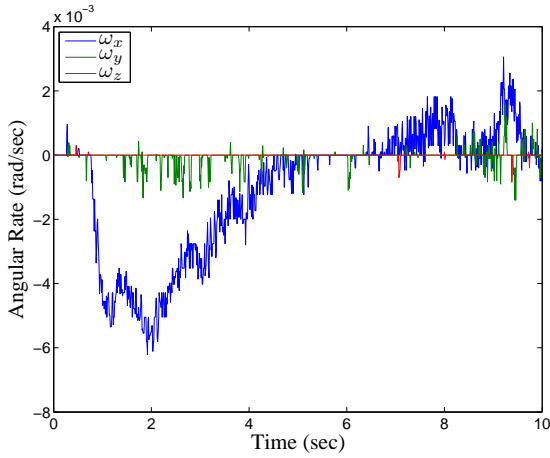


(a) Body Rates

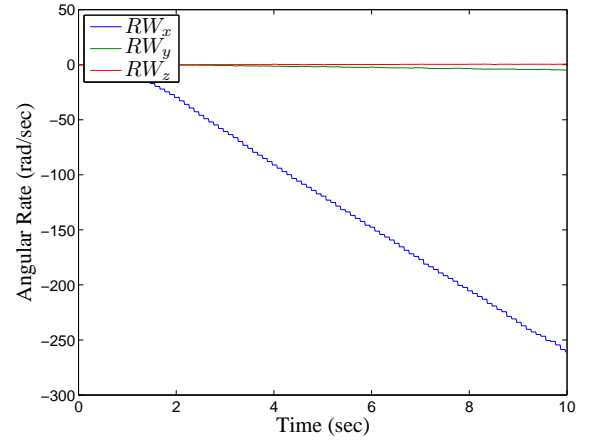


(b) Reaction Wheel Rates Rates

Figure 4.20: Reaction Wheel Disturbance Test X-axis with 17.2 g



(a) Body Rates



(b) Reaction Wheel Rates Rates

Figure 4.21: Reaction Wheel Disturbance Test X-axis with 42.6 g

**4.6.2 CMG Torque Test Results.** The no load control test for the CMG shown in Fig. 4.22 was conducted using the Moore-Penrose Pseudoinverse Steering Law (MPPSL) starting from  $\bar{\delta}_0 = [0, 0, 0, 0]$ . All four gimbals deviated away from zero, storing an angular momentum vector of  $[0.0119, -0.0041, -0.0106]$  N-m-s or a net angular momentum of 0.0164 N-m-s. There was no opportunity for saturation or singularities, therefore the no

load test was not repeated using the alternate initial gimbal angles of  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$  or the Generalized Inverse Steering Law (GISL).

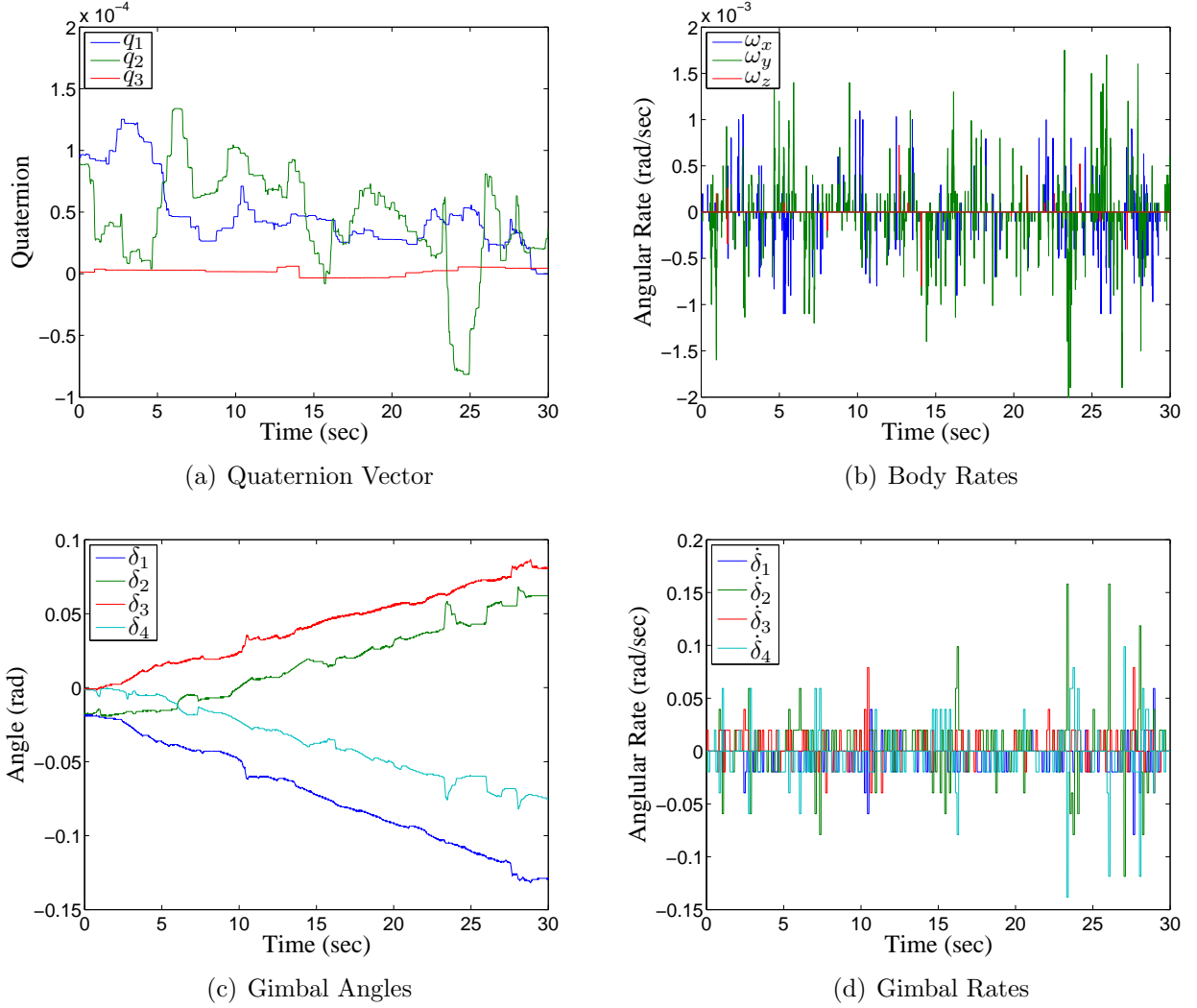


Figure 4.22: CMG Disturbance Control Test with 0 g

**4.6.2.1 X-axis CMG Torque Test Results.** Figure 4.23 shows the results of the 17.2 g X-axis test using the MPPSL with an initial gimbal condition of  $\bar{\delta}_0 = [0, 0, 0, 0]$ . The motion of SimSat indicated in Fig. 4.23 (a) and (b) show that the CMG array stopped producing torque at 6.6 seconds. From Fig. 4.23(c) the angular momentum at the singularity is  $[-0.5193, 0.0150, -0.0085]$  N-m-s or a magnitude of 0.5196 N-m-s, 37% of the theoretical maximum angular momentum available in the X-direction. SimSat was

unable to escape the singularity and the test was terminated after 11 seconds to prevent SimSat from contacting the pedestal.

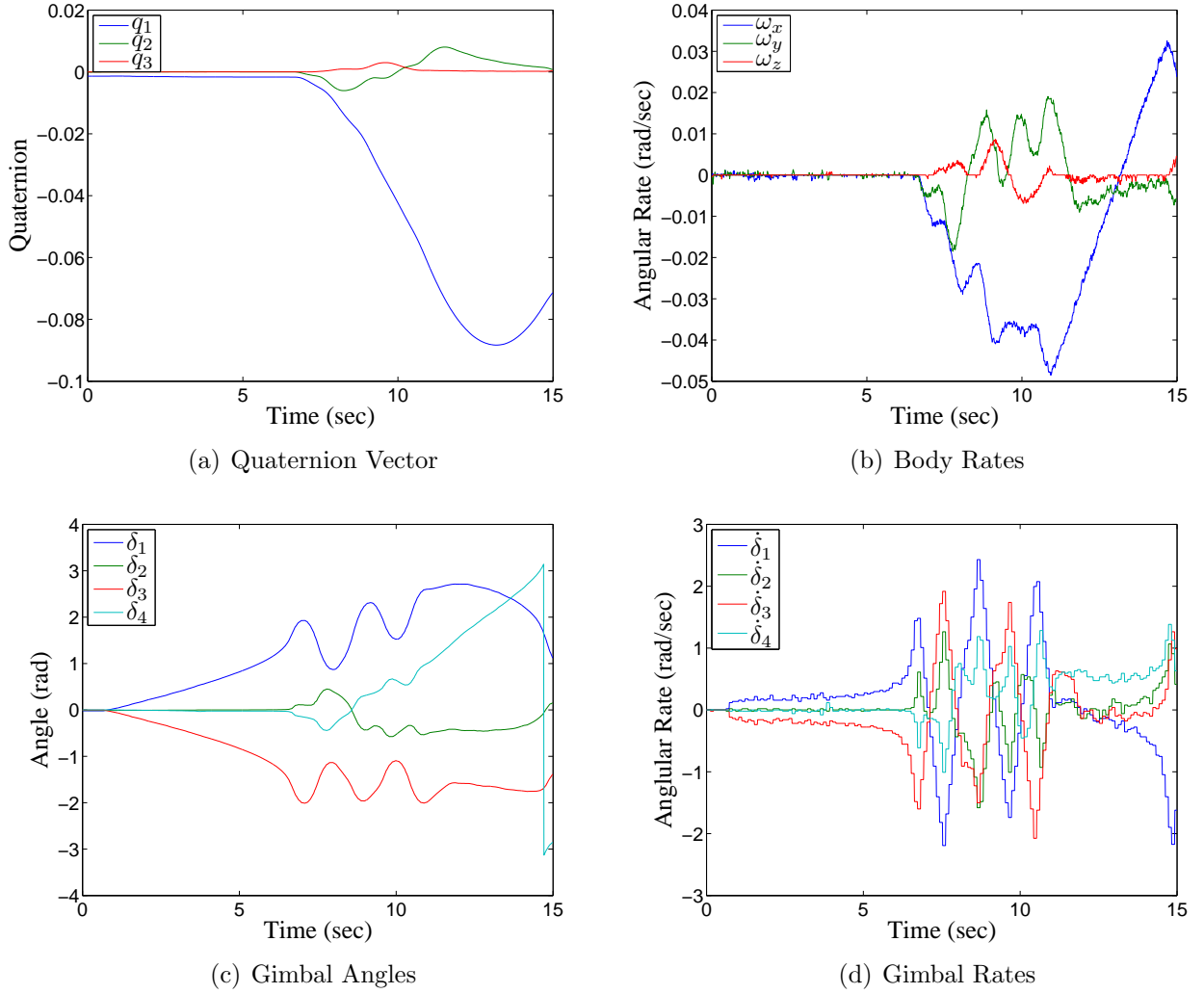
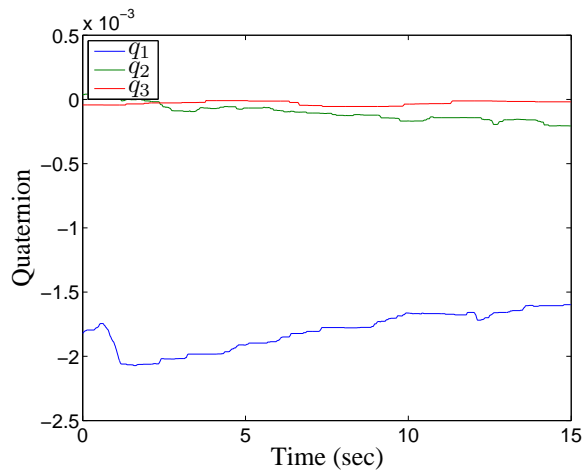
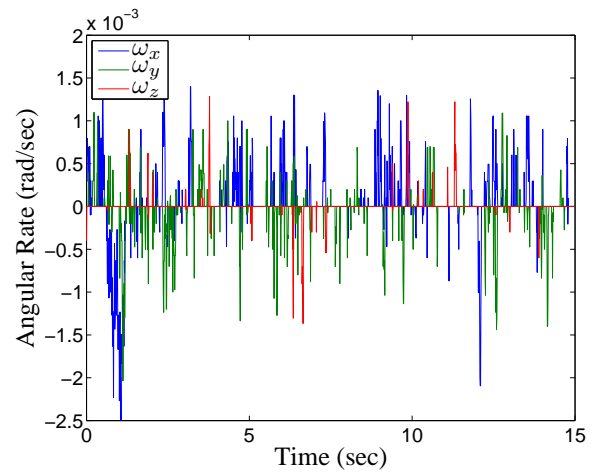


Figure 4.23: CMG Disturbance Test, X-axis MPPSL,  $\bar{\delta}_0 = [0, 0, 0, 0]$ , 17.2 g

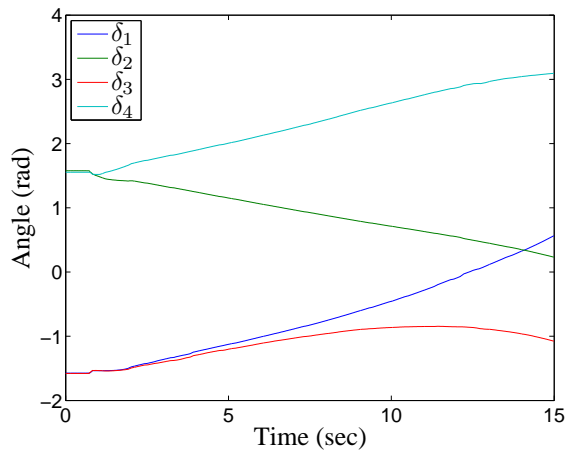
The X-axis 17.2 g test was repeated using the MPPSL with the initial gimbal angles of  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , shown in Fig. 4.24. Unlike the earlier configuration, the CMG array did not reach a singularity during the testing window and was therefore able to hold SimSat within a  $1/4^\circ$  of level, as seen in Fig. 4.24(a). Using the gimbal angles shown in Fig. 4.24(c) the accumulated angular momentum was calculated to be  $[-1.2554, -0.1194, -0.0249]$  N-m-s or a magnitude of 1.2613 N-m-s.



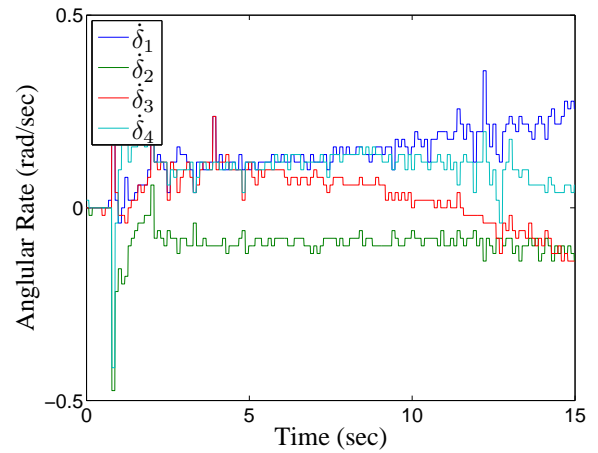
(a) Quaternion Vector



(b) Body Rates



(c) Gimbal Angles



(d) Gimbal Rates

Figure 4.24: CMG Disturbance Test,  $X$ -axis MPPSL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 17.2 g

The GISL test performed on the  $X$ -axis with 17.2g with initial gimbal angles of  $\bar{\delta}_0=[0,0,0,0]$ , shown in Fig 4.25 encountered the same singularity as the MPPSL shown in Fig. 4.23, but Fig. 4.25(a) shows an order of magnitude less deviation, equivalent to  $1.06^\circ$  of total deflection from the desired orientation at 7.8 seconds into the test before recovering. The GISL was able to overcome the singularity by coupling in null motion to move gimbals 2 and 4 away from their initial angles to generate more angular momentum along the  $X$ -direction. For the set of initial conditions tested, the inclusion of null motion improved the overall performance of the CMG array. At the end of the 15 second test, the CMG array has accumulated  $[-1.3030, -0.0679, -0.0048]$  N-m-s or a magnitude of 1.3048 N-m-s.

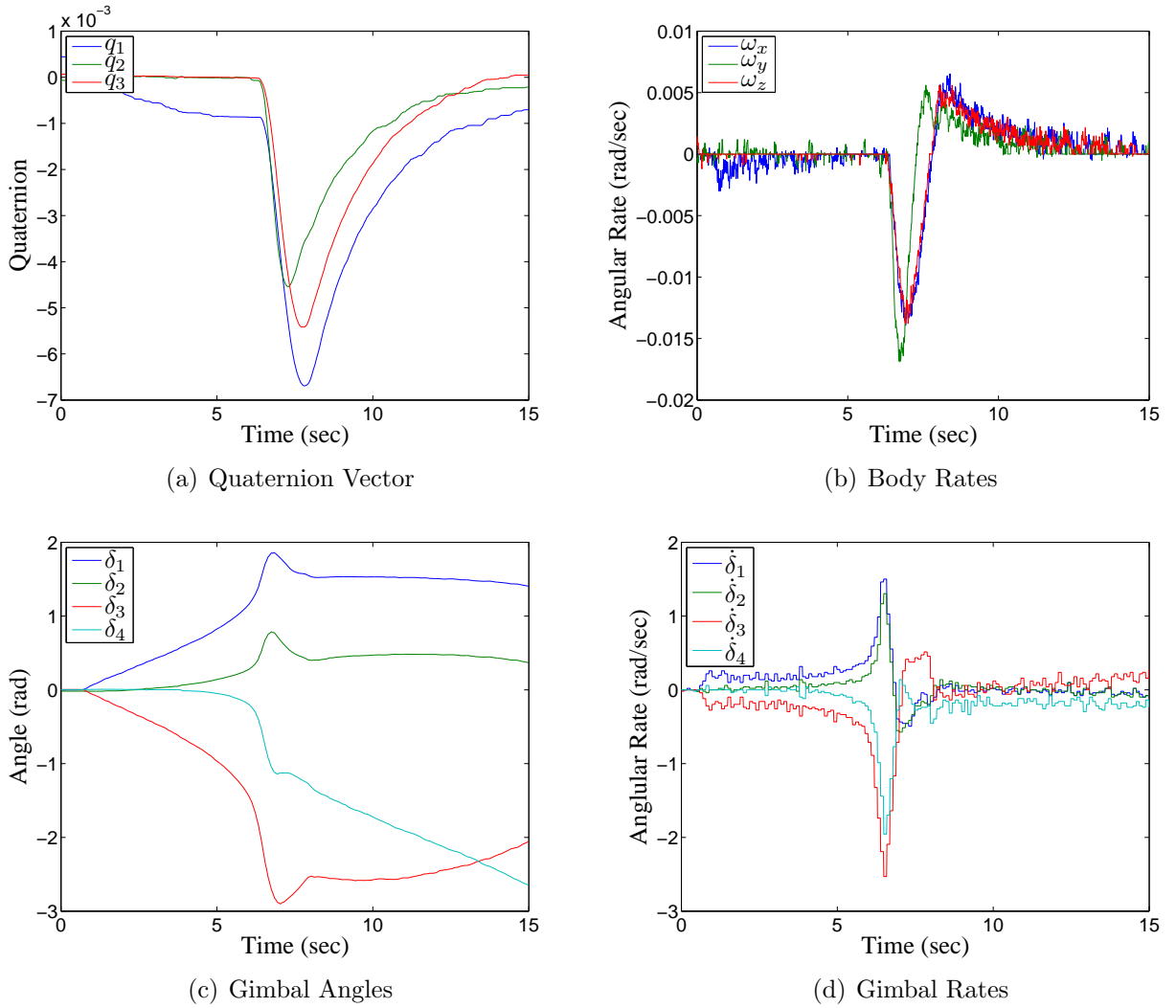


Figure 4.25: CMG Disturbance Test,  $X$ -axis GISL,  $\bar{\delta}_0=[0,0,0,0]$ , 17.2 g

The final test with the 17.2 g applied mass was performed using the GISL algorithm with a gimbal initial orientation of  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , shown in Fig. 4.26. As expected, the GISL had very similar performance to the MPPSL because neither encountered any singularities. The total angular momentum accumulated during this test was  $[-1.2608, -0.1382, -0.0135]$  N-m-s or a magnitude of 1.2684 N-m-s.

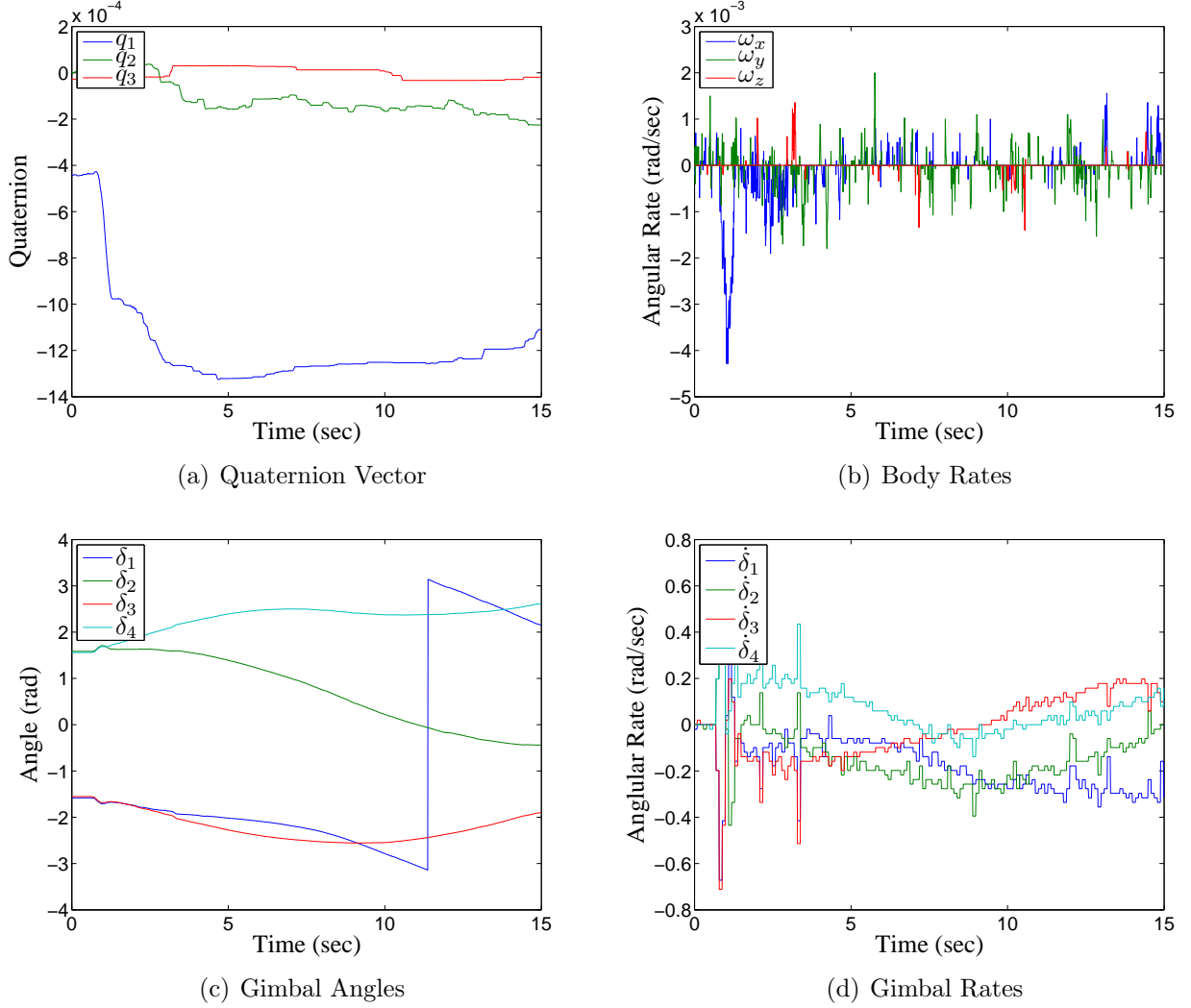


Figure 4.26: CMG Disturbance Test, X-axis GISL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 17.2 g

The X-axis torque tests were repeated with a 42.6 g mass to force the CMG array to saturation. As expected, the MPPSL with initial gimbal angles of  $\bar{\delta}_0 = [0, 0, 0, 0]$ , shown in Figure 4.27, reached the same internal singularity in less 3 seconds from application of the torque. The angular momentum accumulated when the singularity was encountered is  $[-0.3954, 0.0369, -0.0196]$  N-m-s or a magnitude of 0.3976 N-m-s. SimSat was unable



to escape the singularity and was physically stopped around 9 seconds into the test to prevent contact with the pedestal which could alter the position of SimSat's center of mass.

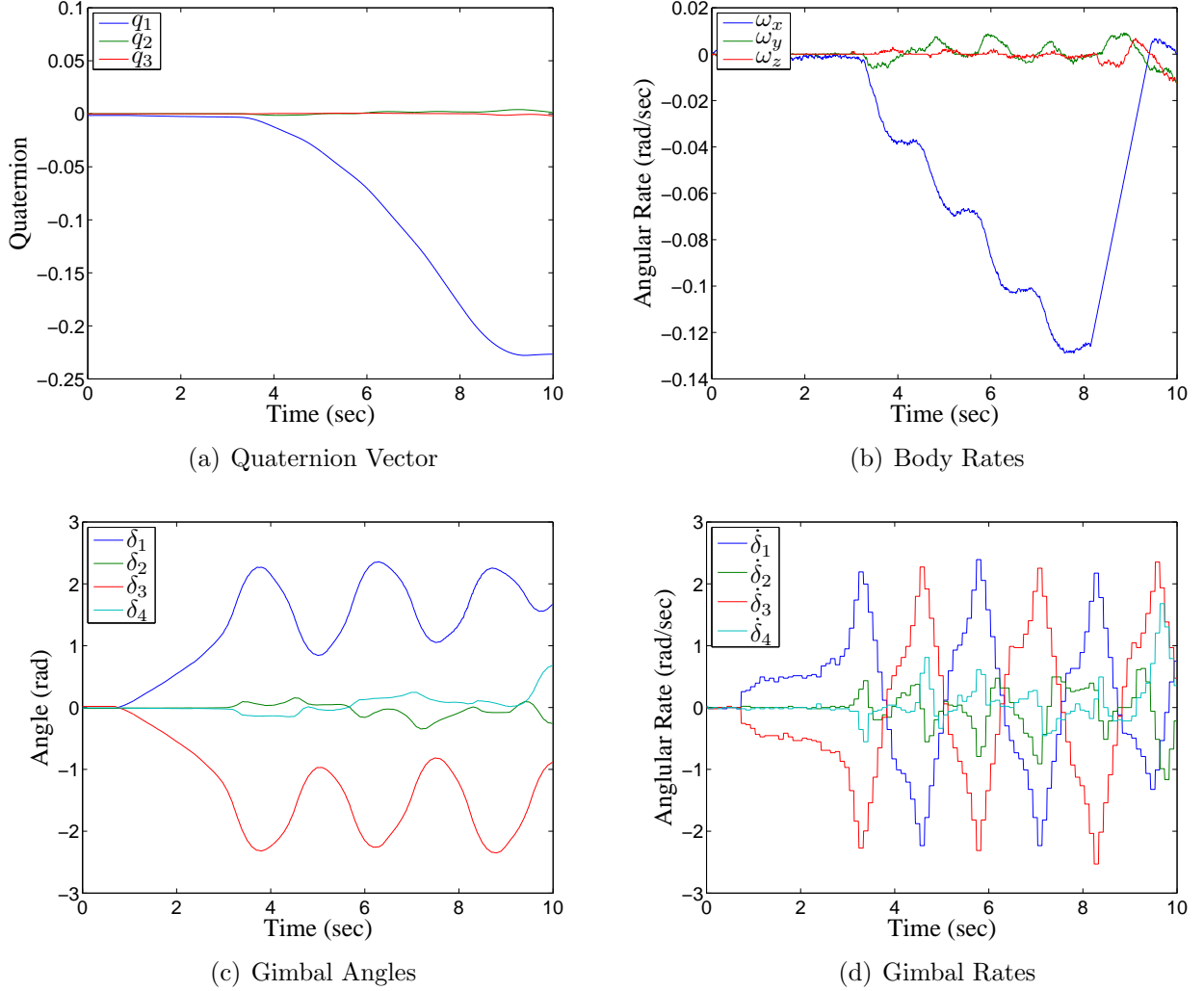
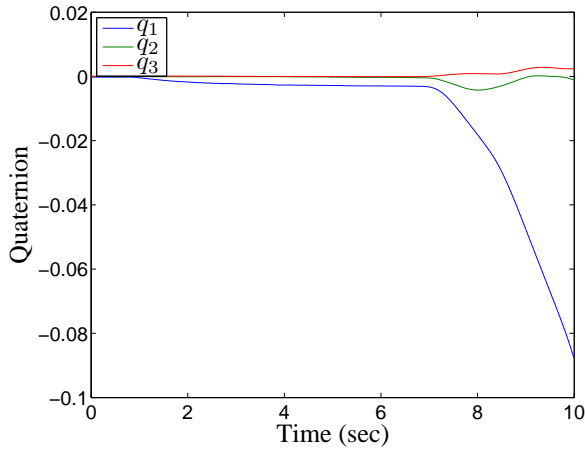
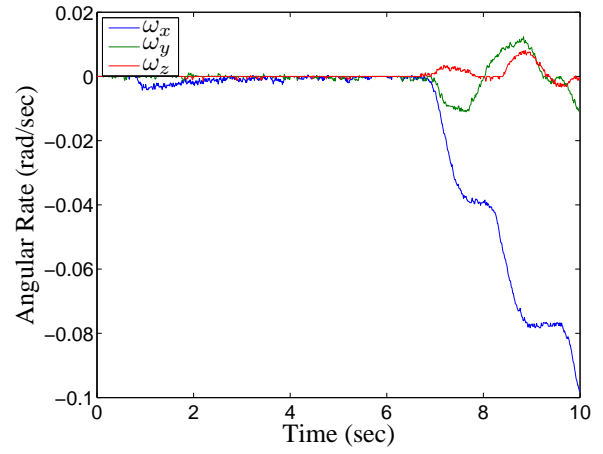


Figure 4.27: CMG Disturbance Test, X-axis MPPSL,  $\bar{\delta}_0 = [0, 0, 0, 0]$ , 42.6 g

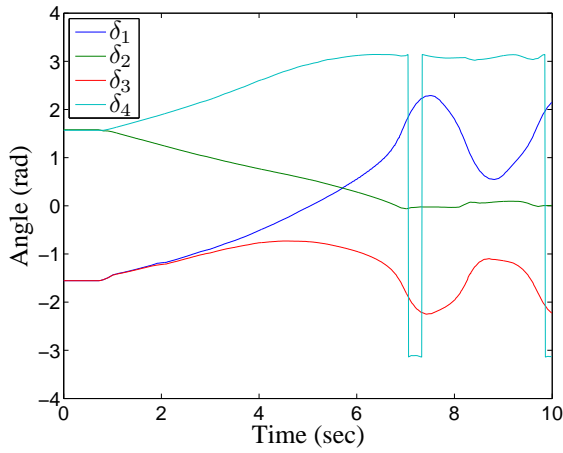
The second test conducted with the 42.6 g mass was performed using the MPPSL with initial gimbal angles of  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , shown in Fig. 4.28. As in the 17.2 g test, the MPPSL performed much more effectively from  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$  than from  $\bar{\delta}_0 = [0, 0, 0, 0]$ , and did not encounter any internal singularities. Figure 4.28 shows that CMG array reached saturation at 7 seconds into the test, with an accumulated angular momentum of  $[-1.4112, -0.0470, -0.0077]$  N-m-s or a magnitude of 1.4120 N-m-s.



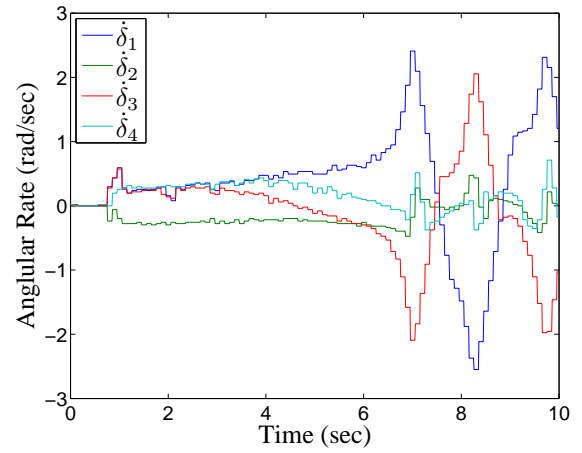
(a) Quaternion Vector



(b) Body Rates



(c) Gimbal Angles



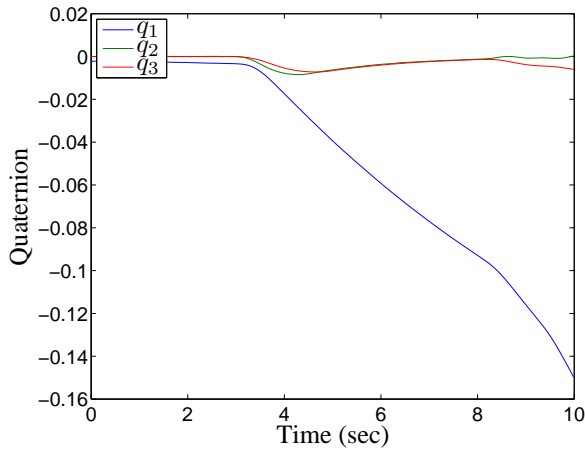
(d) Gimbal Rates

Figure 4.28: CMG Disturbance Test, X-axis MPPSL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 42.6 g

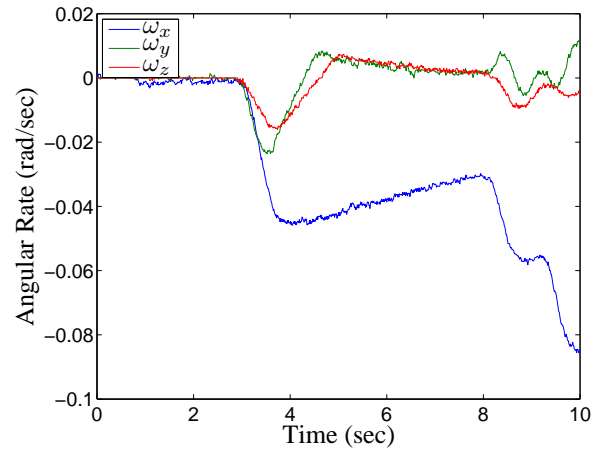
The GISL test performed with the 42.6 g mass and initial gimbal angles  $\bar{\delta}_0=[0,0,0,0]$ , shown in Fig. 4.29, displayed similar behavior to the MPPSL with the same initial angles shown in Fig 4.27 in that it encountered an internal singularity early in the test and began to move away from the desired orientation. In the 17.2 g test case, shown in Fig. 4.25, the GISL was able to use null motion to move around the singularity and maintain enough torque to keep SimSat from moving away from the desired orientation until the singularity had been passed. The null motion can be seen in Fig. 4.25(c) by the movement of gimbals 2 and 4 starting 3 seconds into the test, allowing the CMG array to produce some torque as the array moved around, rather than into the singularity. Figure 4.29 shows that there was enough torque available to slow, but not stop the rotation of SimSat about the  $X$ -axis. The angular momentum of the CMG array when it encountered the internal singularity was  $[-0.5041, 0.0083, 0.0009]$  N-m-s or a magnitude of 0.5042 N-m-s, and the angular momentum at saturation was  $[-1.3894, -0.0406, -0.0235]$  N-m-s or a magnitude of 1.3902 N-m-s.

The final  $X$ -axis test was to apply a 42.6 g mass using the GISL with initial gimbal angles of  $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ . The results are shown in Fig. 4.30. The movement of SimSat away from the initial orientation seen in Figs. 4.30 (a) and (b) indicates that the CMG array does not have sufficient torque to counteract the applied load and couple null motion to avoid singularities. For the combination of  $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$  and 42.6 g of applied load, the addition of null motion reduced the performance of the CMG array without any benefits. At saturation, the CMG array had accumulated  $[-1.4122, -0.0894, -0.0183]$  N-m-s or a magnitude of 1.4152 N-m-s of angular momentum.

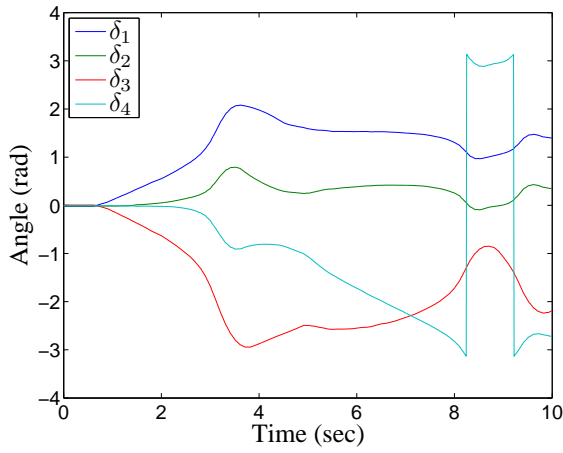
*4.6.2.2 Z-axis CMG Torque Test Results.* As with the  $X$ -axis tests, each control law and initial gimbal angle condition were tested against three applied torques. Figure 4.31 shows the results of the 17.2 g applied mass test with the MPPSL starting with gimbal angles  $\bar{\delta}_0=[0,0,0,0]$ . The CMG array encountered no singularities from the start of the test through saturation 16 seconds into the test. At saturation, the CMG array had accumulated  $[-0.0447, 0.0515, -1.4638]$  N-m-s or a magnitude of 1.4653 N-m-s.



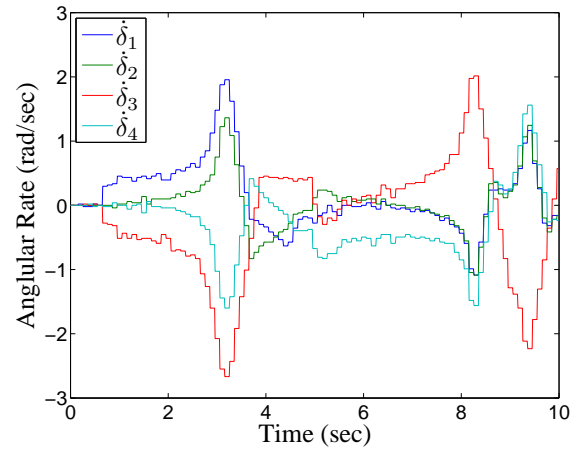
(a) Quaternion Vector



(b) Body Rates

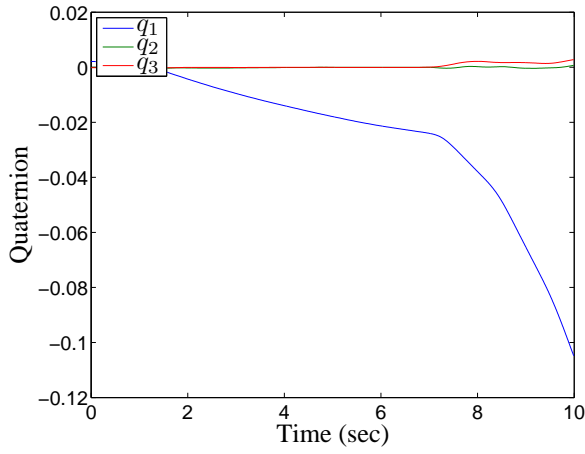


(c) Gimbal Angles

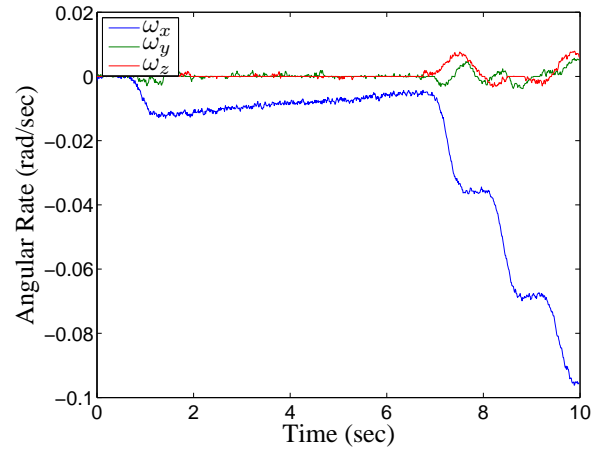


(d) Gimbal Rates

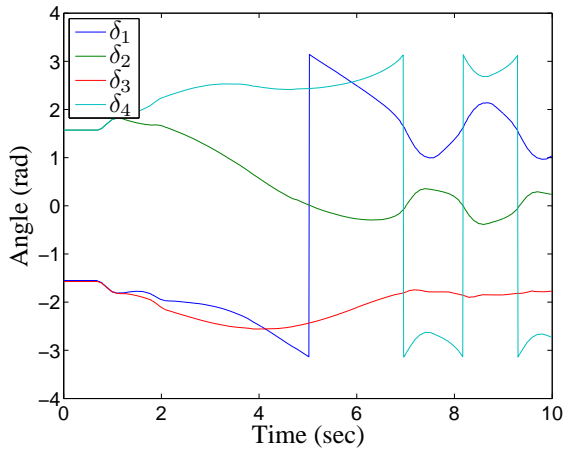
Figure 4.29: CMG Disturbance Test, X-axis GISL,  $\bar{\delta}_0=[0,0,0,0]$ , 42.6 g



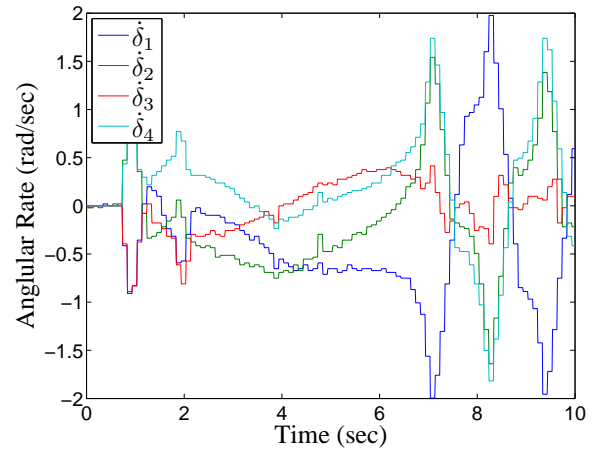
(a) Quaternion Vector



(b) Body Rates

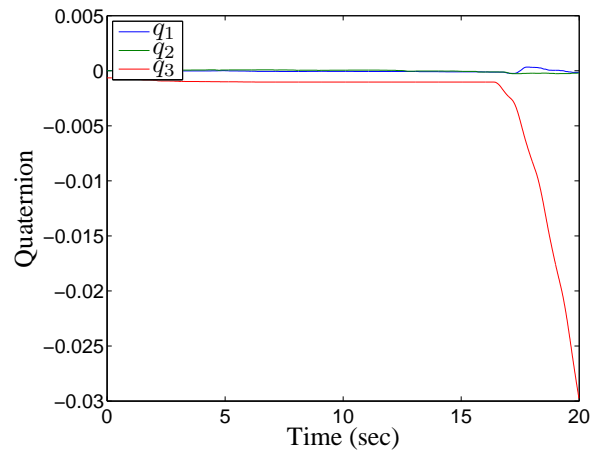


(c) Gimbal Angles

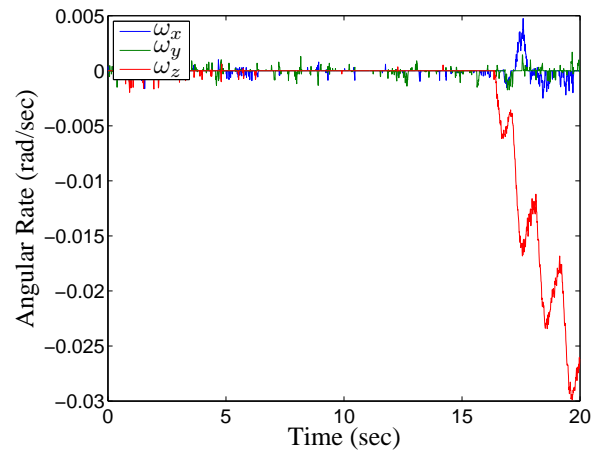


(d) Gimbal Rates

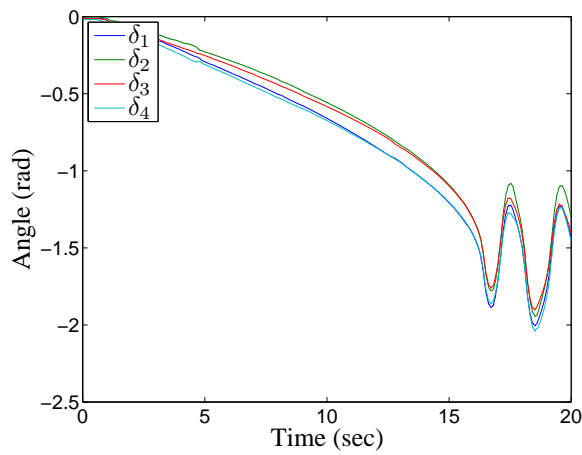
Figure 4.30: CMG Disturbance Test, X-axis GISL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 42.6 g



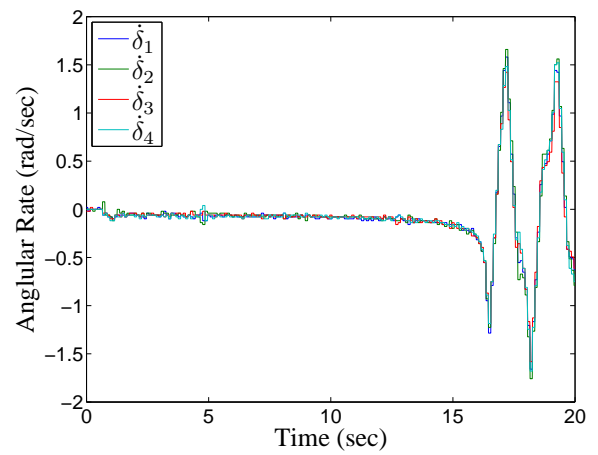
(a) Quaternion Vector



(b) Body Rates



(c) Gimbal Angles



(d) Gimbal Rates

Figure 4.31: CMG Disturbance Test, Z-axis MPPSL,  $\bar{\delta}_0=[0,0,0,0]$ , 17.2 g

The second  $Z$ -axis test at 17.2 g mass was performed using the MPPSL with the initial gimbal angles  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , shown in Fig. 4.32. Figures 4.32 (a) and (b) show indications of an internal singularity about 10 seconds into the test when gimbals 2 and 4 passed through 0 rad. The system passed through the singularity and saturates at 16.75 seconds with an angular momentum of  $[-0.0303, -0.0314, -1.4678]$  N-m-s or a magnitude of 1.4685 N-m-s.

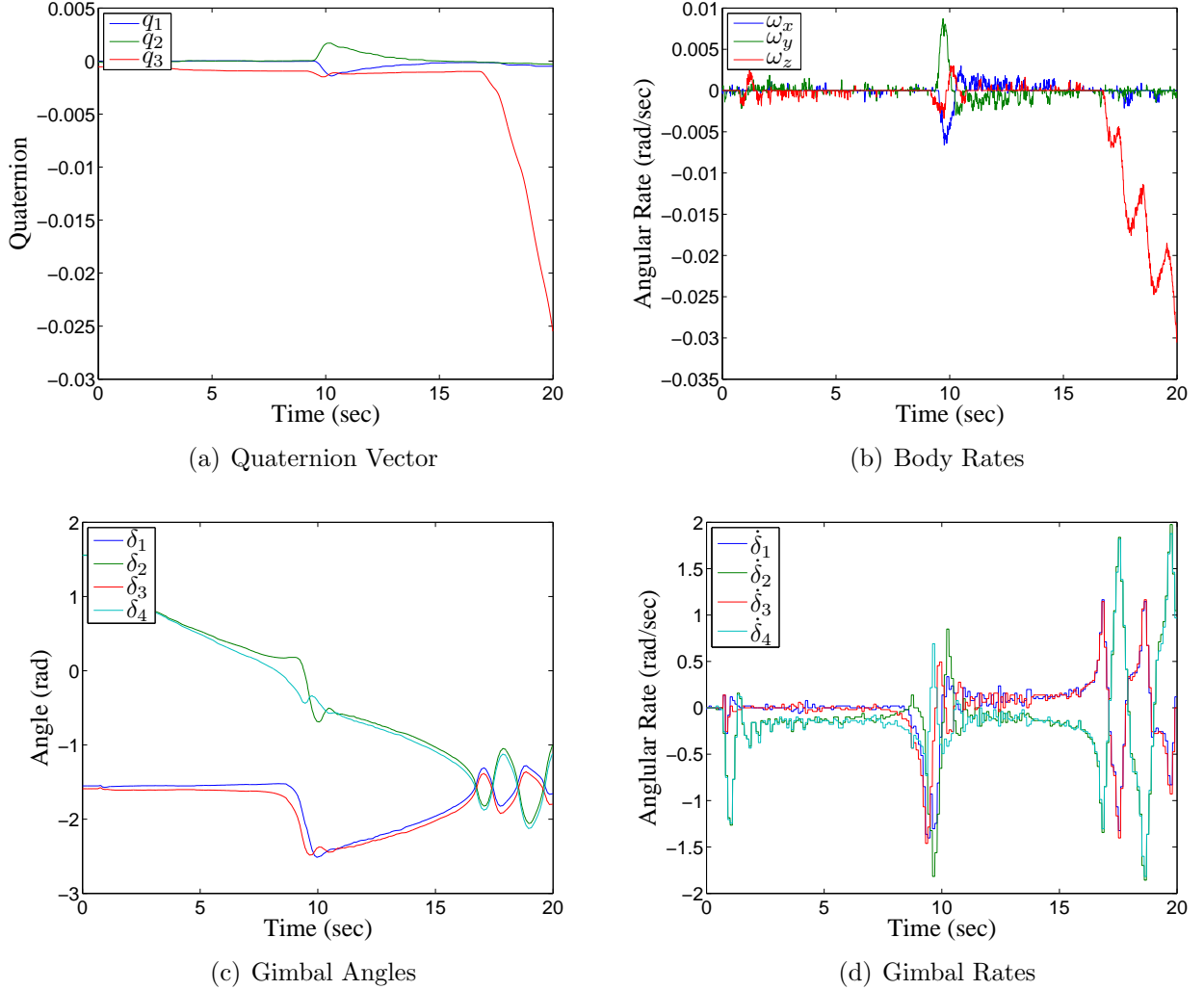
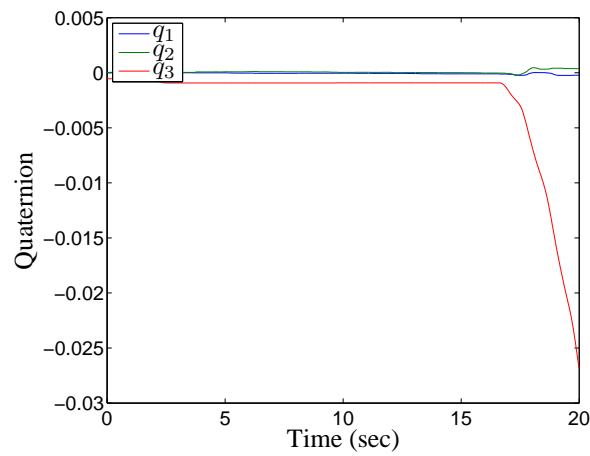
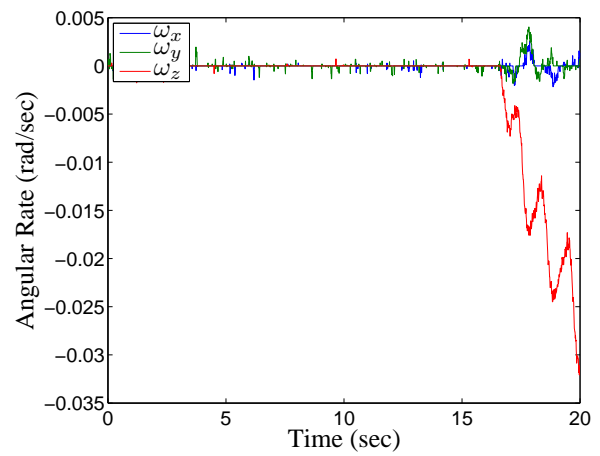


Figure 4.32: CMG Disturbance Test,  $Z$ -axis MPPSL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 17.2 g

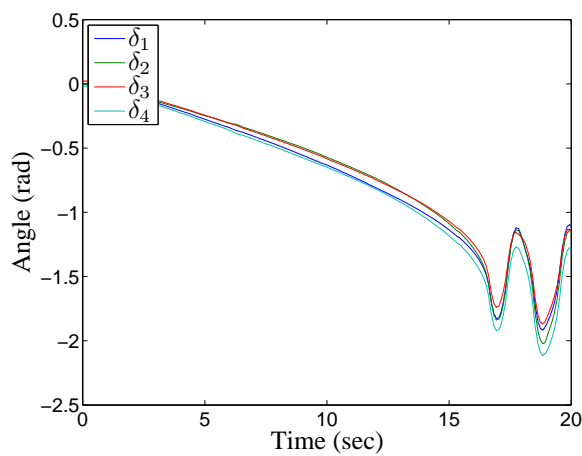
The 17.2 g  $Z$ -axis test using the GISL with initial gimbal angles  $\bar{\delta}_0 = [0, 0, 0, 0]$ , shown in Fig. 4.33, has a response that is almost identical to the MPPSL. The responses are identical because neither case encountered any singularities. The angular momentum at saturation was  $[-0.0386, 0.0282, -1.4637]$  N-m-s or a magnitude of 1.4645 N-m-s.



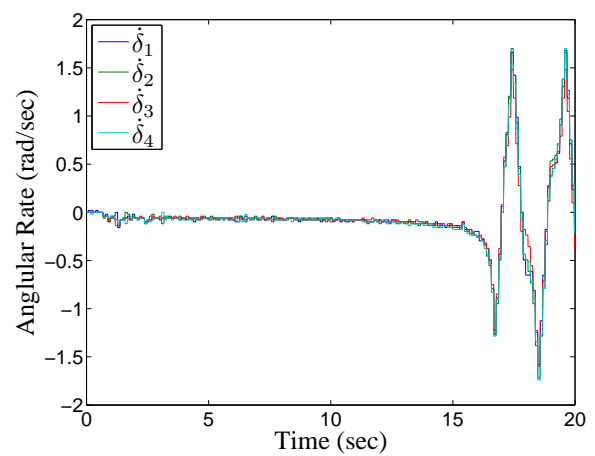
(a) Quaternion Vector



(b) Body Rates



(c) Gimbal Angles



(d) Gimbal Rates

Figure 4.33: CMG Disturbance Test, Z-axis GISL,  $\bar{\delta}_0=[0,0,0,0]$ , 17.2 g



The final Z-axis test performed with a 17.2 g load was with the GISL and initial gimbal angles  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , shown in Fig. 4.34. As with the MPPSL starting with  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , the GISL encountered a singularity almost immediately. Unlike the X-axis GISL test with the  $\bar{\delta}_0 = [0, 0, 0, 0]$ , where SimSat avoids the singularity with minimal deviation, the GISL operating from  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$  was unable to maintain the orientation as it approached and tried to avoid the singularity. The system reached saturation at about 15 seconds into the test with an angular momentum of  $[-0.0422, 0.0024, -1.2923]$  N-m-s or a magnitude of 1.2930 N-m-s.

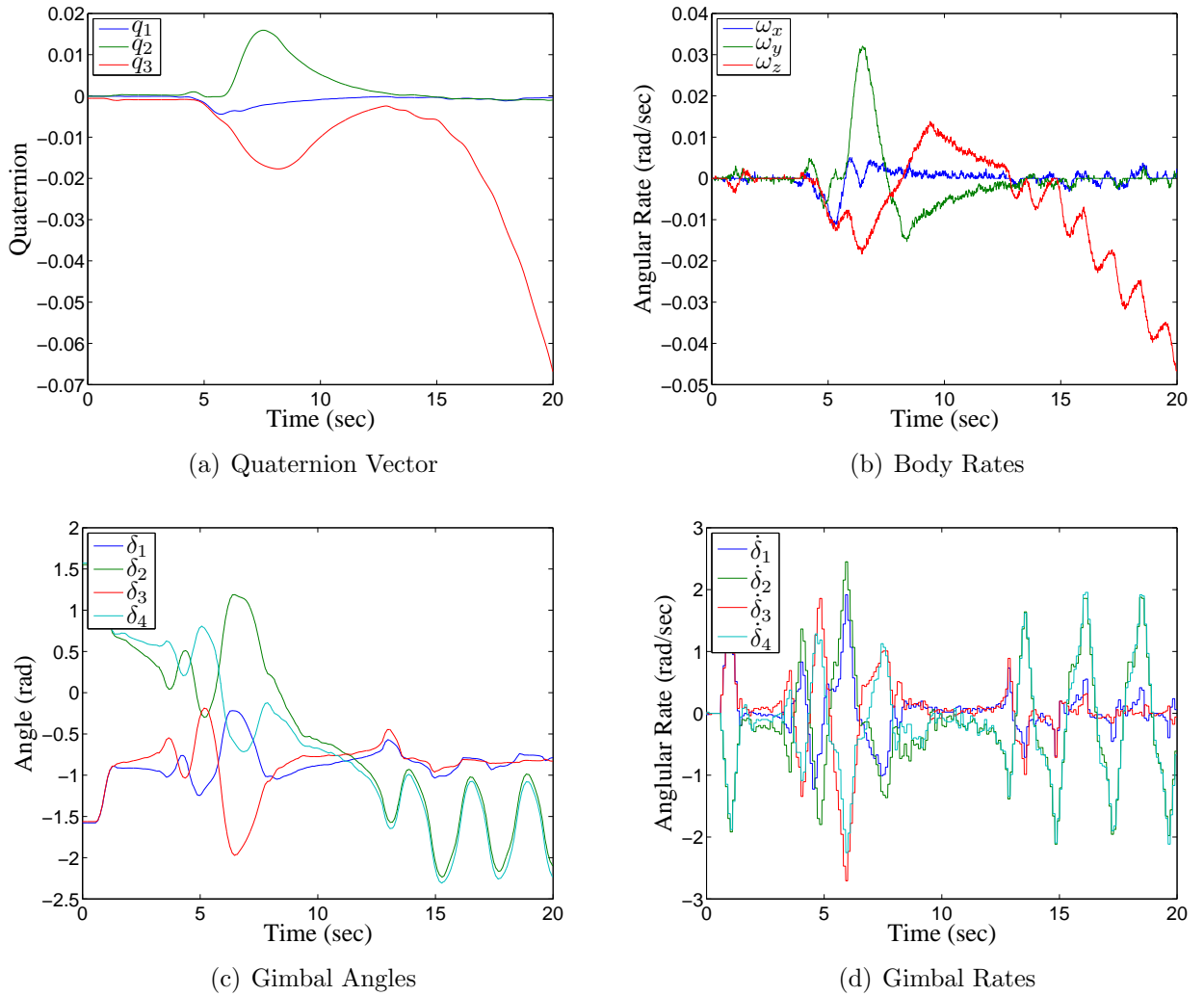


Figure 4.34: CMG Disturbance Test, Z-axis GISL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 17.2 g

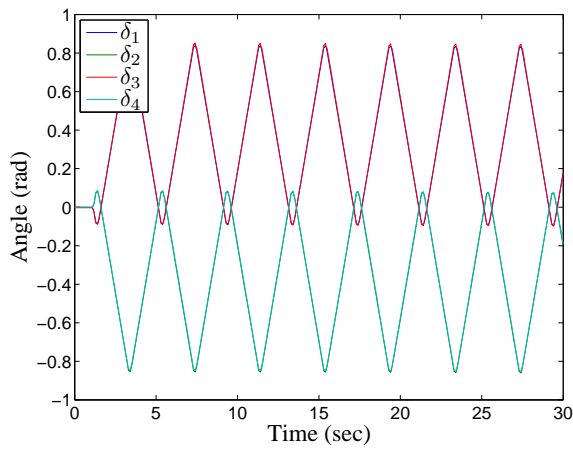
The Z-axis tests were repeated with 42.6 g applied load, but showed no significant deviation from their respective 17.2 g tests except for the reduced time required to reach singularities or saturation. The results of these tests are in Appendix 1.5.2.

#### ***4.7 CMG Null Motion Test Results***

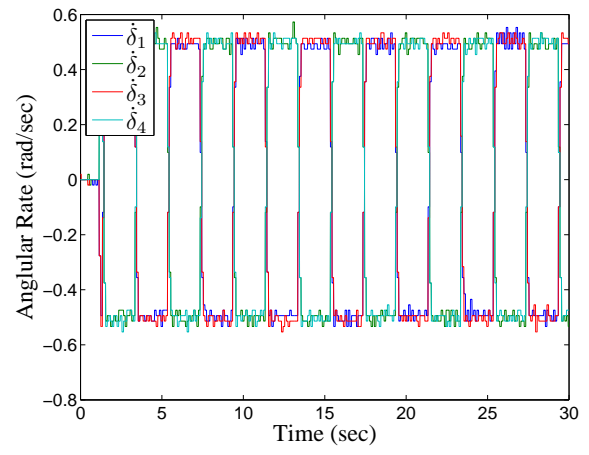
As discussed in Section 3.7.8, the purpose of the null motion test was to characterize the net discrepancies of the CMG array as a unit. As discussed in Section 2.4.1, null motion should result in no change in the CMG array's total angular momentum. Any discrepancies in the CMG design will cause the CMG null motion gimbal movements to change the angular momentum of the CMG array and impart an internal torque on SimSat.

The null motion test was conducted using the reaction wheel ACS to counteract disturbance torques and measure the change in angular momentum of the CMG array. Prior to the null motion experiment, the disturbance torques were measured using the reaction wheel ACS to provide a calibration measurement. Because SimSat remained stationary to within  $\pm 0.5^\circ$  from the start of calibration through the completion of the null motion test the center of mass was assumed to remain constant, allowing for the gravity disturbance torque removed from the null motion test. Figure 4.35 shows the null motion behavior, with Fig 4.35(c) showing the angular momentum accumulated by the reaction wheels after correcting for the disturbance torques measured during the calibration. Figures providing the raw measurements from the null motion test are in Appendix A1.6. From Fig. 4.35(c), it is clear that the null motion does not perfectly maintain zero net angular momentum. There are several potential sources of error:

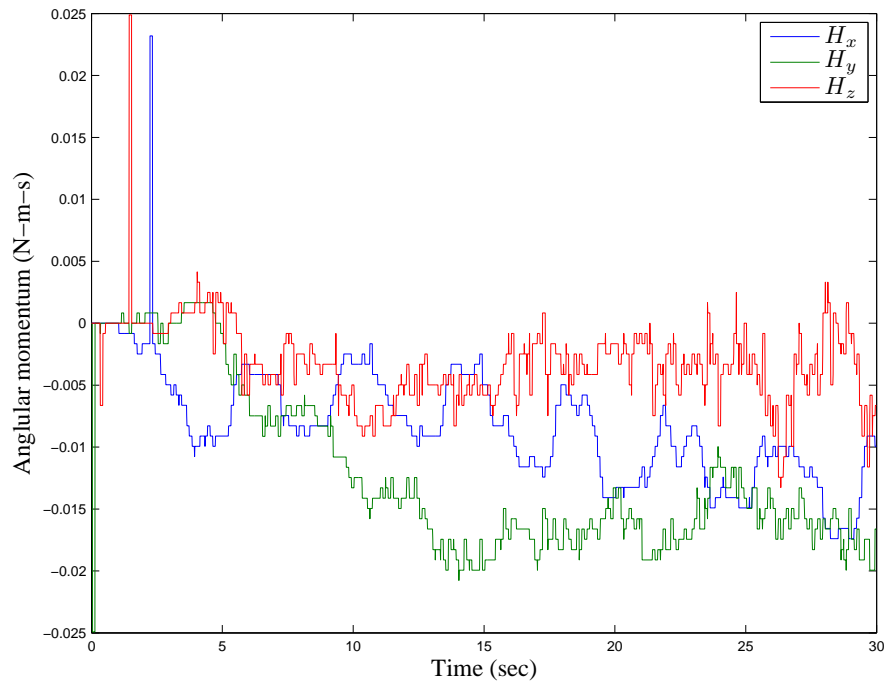
- Unequal angular momentum in each CMG rotor
  - The 1.17% difference between the rotor with the largest MOI and the smallest MOI
  - The potential  $\pm 10$  rpm (0.4%) difference in rotor speeds based on the accuracy of the Cole-Parmer 8210 photo-tachometer
- Unequal angular momenta of the CMG gimbal assemblies



(a) Gimbal Angles



(b) Gimbal Rates



(c) Corrected Angular Momentum

Figure 4.35: Null Motion Test

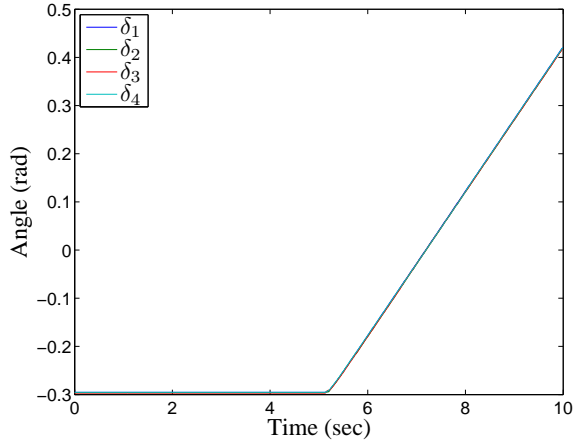
- Gravitational torques caused by the CMG gimbal assembly's center of mass not being aligned with the center of rotation
- Gravitational torques caused by the CMG array moving on its vibration mounts
- Static misalignment of the CMG gimbal axes
- Dynamic misalignment of the CMG gimbal axes due to structural flexing
- External disturbance torques not accounted for in the calibration such as airflow in the room
- Instability in the reaction wheel controller affecting the measurements

The saturation momentum of the CMG array is approximately  $\pm 1.47$  N-m-s about the  $Z$ -axis or  $\pm 1.42$  N-m-s about the  $X$ - and  $Y$ -axes. The total error in null motion is bounded in all three directions by 0.025 N-m-s for a total magnitude of 0.043 N-m-s or 3% of the total momentum available in the  $X$ - and  $Y$ -directions.

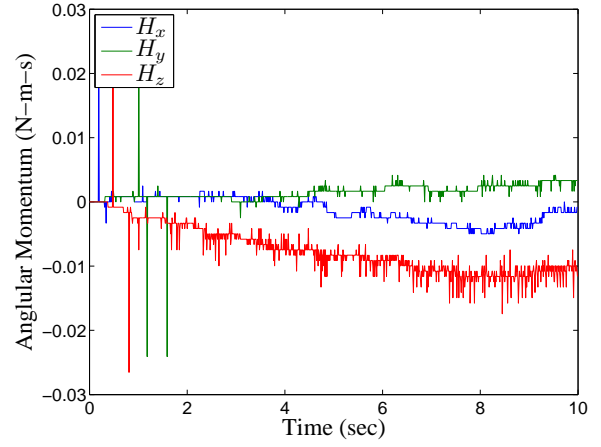
#### ***4.8 CMG Torque Multiplication Test Results***

One of the major advantages of CMGs is that they multiply the motor input torque and thus allow a small input torque to provide a high output torque. The purpose of the CMG torque multiplication test, described in Section 3.7.9, was to measure the internal torques acting on the vehicle and estimate the theoretical and actual torque multiplication of SimSat's CMG array.

The torque multiplication test used the reaction wheel ACS as the torque sensor to measure the torque generated by the CMGs with and without the rotors enabled. As with the null motion test, a calibration measurement was taken prior to measuring the CMG torques to measure and account for the disturbance torques. Figures providing the raw measurements from the torque multiplication test are in Appendix A1.7. Figure 4.36 shows the response to actuating only the gimbals. In theory, only a  $Z$ -axis change in momentum should be measured; however, a disturbance torques about the  $X$ - and  $Y$ -axes was measured during the gimbal test that was not present in the calibration test. These disturbance torques were likely the result of imbalance in the gimbals, flexure in the gimbal assembly, and airflow in the room. The torque effect of the gimbals was calculated

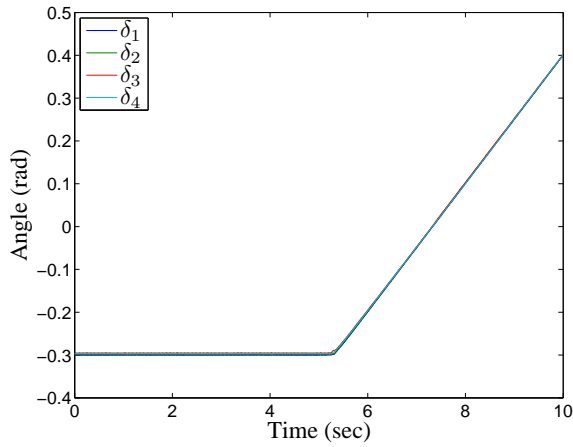


(a) Gimbal Angle

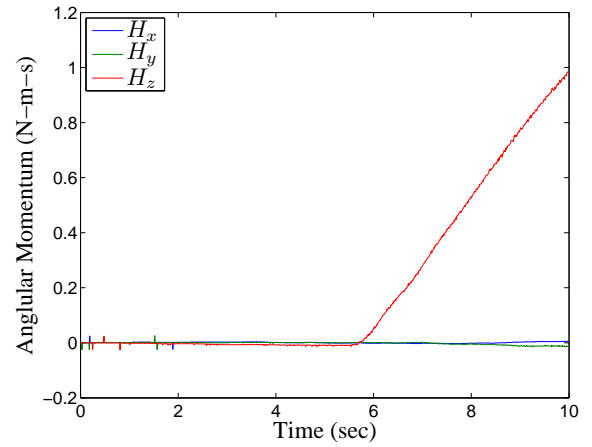


(b) Corrected Angular Momentum

Figure 4.36: Gimbal Torque Effect, Rotors Disabled



(a) Gimbal Angle



(b) Corrected Angular Momentum

Figure 4.37: CMG Torque Effect, Rotors Enabled

by applying a first-order fit to the change in the  $Z$ -axis angular momentum in the 0.5 seconds after the gimbals actuated, then correcting for the geometry and cancellation. The net torque effect produced by a single gimbal was estimated to be  $7.420 \times 10^{-4}$  N-m for a 0 to 0.15 rad/sec gimbal rate command. Figure 4.36(b), shows that the gimbal movement alone imparts almost no torque on SimSat and is nearly undetectable with the disturbance torques. The gimbals stop applying torque to the vehicle once they reach a steady state speed of 0.15 rad/sec; however, motor torque is still required to overcome losses in the gimbal bearings.

The torque produced by the CMG array at 0.15 rad/sec gimbal rate was measured in the same manner. Accounting for geometry, the total torque produced by a single gimbal at 0.15 rad/sec was measured to be 0.06744 N-m. The expected value for the torque produced by the CMG at 0.15 rad/sec is 0.0675 N-m based on the CMG array geometry, rotor angular momentum of 0.45 N-m-s, and gimbal rate of 0.15 rad/sec. The difference between the expected torque and the measured torque of the CMG array is 0.09%. Figure 4.37(b) shows that the CMG torques are significantly larger than the disturbances. Comparing the CMG torque to the gimbal torque yields a torque multiplication factor of about 90.9.

It is important to note that the actual torque required to rotate the gimbal is dominated by bearing friction and transmission losses which were not measured using this method. The actual gimbal torque can be estimated using the motor's 'torque constant,' which relates torque to current draw and is an average value for the motor, and the current draw. The EC-MAX-30 motor has a torque constant of 35.9 mN-m/A and an average current draw of per motor 75 mA when operating at 0.15 rad/sec resulting in an estimated torque of  $2.7 \times 10^{-3}$  N-m. Using the electrical estimate, the torque multiplication factor is 25, indicating that the vast majority of the torque required to rotate the gimbals is lost to friction in the transmission and bearings, which is acceptable on SimSat.

As a comparison, the total current draw of the CMG rotors while operating is 0.7 A continuous draw measured when operating on ground power. At 0.22 N-m of torque, the gimbal motors each require 0.075 A of power to operate, for a total CMG

power consumption of 1 A. The  $Z$ -axis reaction wheel consumes 7.4 A continuous current and has a transient draw of nearly 12 A when generating the same amount of torque.

#### **4.9 Summary**

Chapter IV presented the results and analysis of the system verification and performance tests. The system verification tests were conducted to validate that the reaction wheels and CMG array are performing to design. There are two exceptions; first, the reaction wheel actuators have a small but significant phase lag in the velocity controller, and second, the reaction wheel angular rate saturation limits are limited by the electrical current available. Methods for addressing both issues are presented in Section 5.2.1. All of the verification tests performed on the CMG array indicate that it is performing within expectations, to include having minimal deviation from null motion and a reasonable torque multiplication effect.

The performance testing shows that SimSat is capable of meeting the stated performance specifications, and the results indicate that the PID controller gains need to be tuned to meet the desired slew maneuver performance specifications. Additionally, integrator windup within the PID caused measurable performance issues with respect to the settling time specifications. The disturbance torque rejection tests demonstrated that both the reaction wheels and CMGs could address disturbance torques up to their control limit. The disturbance torque tests also demonstrated some of the complexities associated with CMGs due to singularities and the impact of not having singularity avoidance steering laws.

## V. Conclusions and Recommendations

### 5.1 Conclusions

The primary objectives of this research effort were to design and build a CMG array for AFIT's SimSat, integrate it with the existing ACS, and validate the performance of the ACS using the CMG array against the requirements listed in Section 1.2. The secondary objectives were to improve the existing reaction wheel ACS subsystems by addressing known issues with system timing, increase angular momentum storage, and provide protection against electrical feedback. The final result of this research is an operational satellite simulator equipped with fan/thruster simulators, reaction wheels, and a 4-unit pyramid array of single gimbal CMGs that can function in concert with one another to provide attitude control. The thesis addresses the system dynamics, hardware/software interfaces and performance characteristics of SimSat in its current configuration. This document also covers the problem solving process used to address known problems, the design process for developing new hardware, and the methodologies for construction and testing.

The research methodology was developed to address the primary and secondary objectives in tandem whenever possible. The most notable secondary objectives addressed early in the research were the actuator timing issues, which have adversely affected the second generation SimSat since it was first declared initial operating capable.[21, 34] The complexity inherent in operating CMGs demanded that the timing problem be solved to ensure SimSat could operate the CMG array once it was ready. The timing issues were solved by altering the actuator communications protocols and adding a TDMA communications scheme.

Concurrent with addressing the timing issues, research on the mechanics, dynamics, configurations, and behaviors of CMG arrays was conducted to determine the system level requirements and establish a tradespace for the CMG arrays. After narrowing down the options, the pyramidal CMG array configuration was selected because of its uniform momentum envelope and physical arrangement which worked well with physical constraints imposed by SimSat's existing design. Next, the critical parameters of the CMG system were scaled to meet system requirements and constraints. These design param-



eters were used to select of commercial components and design the CMG rotor. The remaining elements of the CMG array were then designed around the pyramidal configuration, rotor design, and commercial hardware. Finally, SimSat was fully disassembled and hardware was relocated to provide space for the CMG array. Addressing the reaction wheel electrical issues was handled as part of the rebuild process.

The installation of CMGs required that significant modifications be made to the SIMULINK control program. This was used as an opportunity to implement solutions to the actuator timing issues and address the known singularities in the existing control logic. A full rebuild of the code was completed, using the best elements of previous research addressing as many of the known problems as possible. The control program was then used to conduct verification and validation tests on the new reaction wheels and CMG array. The control program was also used to measure SimSat's mass properties. Included in this testing was verification of properties that are unique to CMG arrays, specifically null motion behaviors and torque multiplication effects. This initial testing verified that the CMG array was functioning as designed and the reaction wheels were functional but have torque output and momentum storage below the design specifications. Recommendations for addressing the reaction wheel issues in Section 5.2.1.

The ACS was then tested using both the reaction wheels and CMGs against the performance specifications listed in Section 1.2. Both systems demonstrated the ability to maintain a fixed attitude to within  $\pm 0.01^\circ$  while rejecting nominal disturbance torques. The reaction wheel and CMG ACSs were also able to meet the accuracy requirements of rest-to-rest maneuvers of  $10^\circ$  about the  $X$ - and  $Y$ -axes, and a  $30^\circ$  maneuver about the  $Z$ -axis. Neither ACS system was able to meet the settling time requirement for rest-to-rest maneuvering with the controller settings currently in use. Based on the rise time time of ACS configurations, addressing the settling time should be matter of adjusting the PID gains to the appropriate level and mitigating the effects of integrator windup and nonlinear control saturation. Addressing integrator windup is discussed later in Section 5.2.2.

The ACS was also tested against applied disturbance torques, and it was during this testing that the complexities of CMGs became readily apparent. The reaction wheels

on SimSat work independently to each provide a specific, decoupled component of the control torque. The CMG must operate together as a unified array to produce the desired control torque, but due to geometry are restricted from moving in certain directions. The disturbance torque testing of the Moore-Penrose Pseudoinverse Steering Law and Generalized Inverse Steering Law demonstrated the effects of steering law decisions on CMG performance. It was clear that each steering law has advantages and disadvantages. These tests also demonstrated that singularities were dependent not only on the static configuration of the gimbals, but the dynamic gimbal angles. Applying the same torque with the same steering law and starting from zero net angular momentum, but with a different set of gimbal angles, resulted in vastly different behavior for both steering laws. The complexity of CMGs is offset by the advantages they offer, specifically high torque and/or large angular momentum storage for a reasonable amount of size and electrical power. The testing of SimSat’s CMG array demonstrates the potential advantages of CMG, if the issues related to singularities can be addressed.

In conclusion, the hardware and software developed in this thesis research has improved AFIT’s satellite simulator by addressing many known deficiencies, improving the existing hardware and software, and providing a new type of actuator for attitude control experimentation. Overall, this research and the hardware it produced provides AFIT with a unique capability to perform future research in the field of spacecraft attitude control.

## ***5.2 Recommendations for Future Development***

Below are the areas of improvement for SimSat and suggested areas of research that SimSat can support.

*5.2.1 Vehicle Hardware Improvements.* The CMG array built for SimSat in this research is operating at just under 45% capability because the CMG rotor motors are only commanded to 4.5 V, rather than their full 10 V range. Because the MicroAutoBox’s DAC can only produce 4.5 V, a signal amplifier should be designed to map the MicroAutoBox’s 0-4.5 V signal to a 0-10 V range. The addition of a signal amplifier

would allow the MicroAutoBox to command the rotor motors to their full speed range specified as 6000 rpm no load speed. Correcting for the bearing loads, the rotor's maximum angular rate should be 5800rpm or better, increasing angular momentum storage of the each CMG from 0.45 N-m-s to 1.00 N-m-s. Increasing the angular momentum of the rotors will also increase the torque multiplication factor, allowing for reduced gimbal rates for the same torque, or higher torque for the same gimbal rates provided the controller saturation limits are relaxed.

The combination of the current limiting resistor circuit and the 20 cm reaction wheels has created a situation where the EPOS motor controller can attempt to draw current in excess of the 13.7 A available. When the EPOS attempts to draw more current than available, a low voltage fault occurs and the EPOS must be reset. As noted in Section 3.7.2, the electrical current limit is restricting the EPOS controller gains which in turn limits the ability of the EPOS to track velocity inputs and creates a phase lag in the velocity controller. The electrical current limitation also restricts the saturation envelope of the reaction wheels. Increasing the current limit can be done by decreasing the resistance of the current limiting resistor array. The batteries have a nominal current limit of 30 A and operate at up to 42 V when fully charged. A minimum of 5 A should be reserved for the SimSat's computers, therefore the current limit for the reaction wheels should be no higher than 25 A, giving a resistor value of 1.6  $\Omega$  with the appropriate heat dissipation.

SimSat is not a rigid body and as a result its center of mass shifts during operation as wiring and components flex, moving the center of mass away from the center of rotation and causing a gravity disturbance torque. The gravity disturbance torque was by far the most significant source of uncertainty and adversely affected every experiment. The original goal was to adjust SimSat's center of mass location to provide neutral stability; however, eliminating wire movement and structural flexing proved to be impossible in practice, therefore an alternate solution is required. One option is use a set of active counterweights to keep the center of mass coincident with the center of rotation and counteract any flexing. Care must be taken in the design of the counterweights to ensure that their motion does not create additional disturbance torques.

SimSat's reaction wheels are exposed to the air resulting in significant drag when the wheels approach saturation. The resulting airflow around the wheel imparts a disturbance torque on the vehicle, which must be counteracted by increasing the wheel speed. A housing around the reaction wheels should be designed that evenly distributes the airflow minimize the disturbance torque.

The extension of SimSat's fan/thrusters to clear the reaction wheels decreased the stiffness of their mounting hardware, allowing the fans to vibrate during operation. Additionally, the fan shrouds are not rigid and would twist during operation which ultimately changes the direction of the torque produced by the fans/thrusters. Both of these issues should be addressed if the fan/thrusters are to be used for precision control.

SimSat's attitude determination system is limited to only the LN-200 IMU and is unable to independently determine its attitude in inertial space. The lack of external references requires the operator to manually align SimSat before most tests. Additionally, the LN-200, like all strap down inertial platforms, drifts over time as sensor and timing errors accumulate, limiting the length of experiments. Adding external reference sensors, such as star tracker, would address both issues and allow SimSat to independently determine and maintain its orientation within the laboratory for extended periods of time.

On several occasions, SimSat's batteries died without warning, resulting in data loss, including the control program, and risking damage to the on-board electronics. The time required to fully discharge the battery is dependent motor operation, and varies from experiment to experiment. The lithium-polymer battery packs that power SimSat have an approximately linear relationship between voltage and capacity from 42 V when fully charged to 34 V when nearly discharged. A battery monitoring circuit should be installed that interfaces with the inputs on the MicroAutoBox to warn the operator and prevent data loss.

Finally, if the CMG's performance envelope requires further expansion, replacing the current rotors with a denser material would be the easiest approach. Tungsten-copper alloys offer up to twice the density of stainless steel, are machinable, and can tolerate the stress causes by high angular rates. By comparison, the rotor motors currently installed

are the highest speed motors that Maxon makes for the torque and power requirements of SimSat’s CMG.

*5.2.2 Vehicle Software Improvements.* SimSat’s attitude determination subsystem operates using the LN-200’s measurements with minimal filtering and accepts a significant amount of noise and does not account for the vehicle dynamics. As a result, there are often measurement spikes, what McFarland referred to as “isolated gyro corruption” where the LN-200 would indicate an angular velocity change far in excess of SimSat’s capabilities. One specific consequence of the signal noise is accelerated measurement drift. A state estimation algorithm that accounts for the SimSat’s dynamics and control inputs, such as the Kalman filter, would address the noise and improve the overall accuracy of the ACS. A Kalman filter may also be useful in limiting the noise measured in the CMG gimbal rate measurement, although the gimbal rate measurement noise did not present a significant issue in this research.[36]

There are several deficiencies in the linearized PID control that should be addressed in future research. The first is the nonlinear effects of control saturation and integrator windup. The saturation limited integrator used in this research mitigated, but did not eliminate the adverse impacts of windup. An alternative that showed promise in simulation was a hybrid PD-PID control algorithm, where the integrator component is only enabled if the absolute error signal is less than the controller saturation limit divided by  $K_P$ . The hybrid PD-PID controller should provide the steady-state performance of a PID controller but have a rise and settling time behavior much closer to a PD control. Additionally, the gains currently used in the controller were tuned in an ad-hoc manner and are not optimal and should be addressed for future research. Finally, the PID algorithm lacks a body rate control mode and can only perform position hold and rest-to-rest maneuvers. A rate control mode would be useful for demonstrating vehicle dynamics and analyzing cross coupling behaviors related to rotation about non-principal axes.

The reaction wheel and CMG actuators have a finite momentum storage and require periodic momentum dumping. The fan/thruster system was successfully used for

manual momentum dumping, and an automatic momentum dumping algorithm was incorporated in the SIMULINK program used prior to the installation of the CMGs. Changes to the reaction wheel command algorithms prevented the existing momentum dumping algorithm from being ported to the current program. A new algorithm for momentum dumping for the reaction wheels and CMGs should be developed to allow for long duration experiments.

MATLAB SIMULINK and the dSpace toolboxes are all significantly out of date, which limited the ability to develop and validate simulations and algorithms without working directly on SimSat. Ideally, SimSat's system dynamic could be built in a SIMULINK model for development on other computers, and then the control elements copied over to SimSat for hardware-in-the-loop testing. Currently, porting models from SIMULINK R2010 (current version) to SIMULINK 6.2 (SimSat version) requires rebuilding the model by hand, rather than simply copying the blocks from one model to another. Additionally, the most recent versions of MATLAB and SIMULINK have improved development environments which reduce the time and effort required to develop control programs. If SimSat is to support multiple simultaneous research efforts, then the issue of SIMULINK compatibility must be addressed.

*5.2.3 Research Areas.* The tests used to measure SimSat's MOI met with mixed success due to disturbance torques and the false assumption that the  $X$ -,  $Y$ -, and  $Z$ -axes were the principal axes. Experimental results indicate that the products of inertia are not zero and can not be ignored. One solution is to attempt to relocate components and re-balance SimSat. Another option is to accept that the defined axes are not the principal axes and determine the full MOI matrix for the currently defined body axes. The full MOI matrix could then be incorporated into the nonlinear correction subroutine to minimize the impact of the products of inertia on the system behavior.

The CMG array's vibration isolation system was not tested or tuned during this research. The vibrations generated by the CMG array were not large enough to interfere with the LN-200, but should still be characterized in order to fully validate SimSat. The vibration isolators were designed to be adjustable, and a full vibration analysis

should be performed to determine the appropriate polyurethane bushing and compression combination that minimize vibration transfer to the vehicle.

The fan/thruster subsystem was not accurately re-calibrated to account for the change in moment arm caused by their relocation. Additionally, previous efforts by McFarland met with mixed success.[21] The reaction wheel subsystem was successfully used to estimate CMG gimbal torques on SimSat by having the reaction wheel ACS maintain position while the gimbals provided a ‘disturbance’ torque. The same technique should be applicable in mapping a more accurate torque-to-RPM curve for the fans/thrusters. The addition of a rate control mode to the linearized controller would allow the fan/thruster mapping to include both static and dynamic thrust.

Finally, the primary goal of this research was to provide a hardware-in-the-loop platform for attitude control experiments with CMGs. The next logical step is to implement additional steering laws on SimSat, such as real-time optimal control or hybrid control/steering laws, to validate their behavior on real hardware. Other research opportunities include studying the effects unequal or unsteady control lag on CMG stability and study the performance and behavior of unequal momentum CMG configurations.

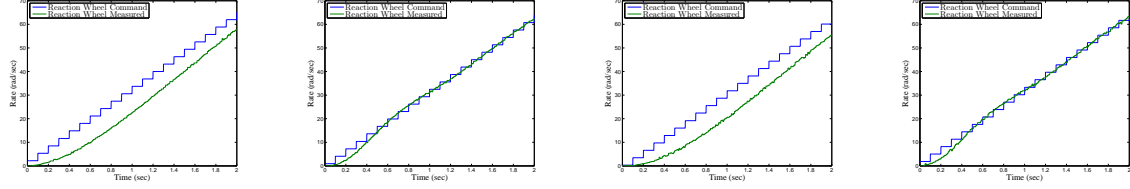
# Appendices



## Appendix A. Supplemental Results Figures

Appendix A contains supplemental results from the tests presented in Chapter IV.

### 1.1 Reaction Wheel Calibration Results



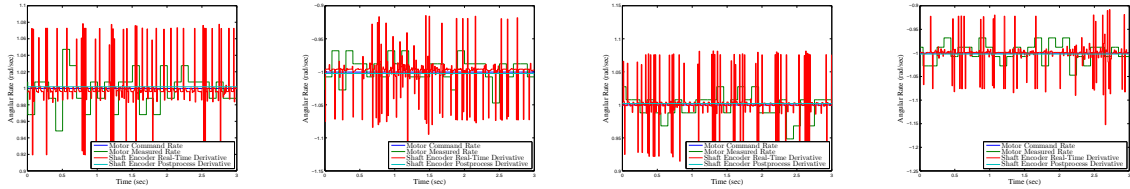
(a) Y-axis  $K_P = 15,000$  &  $K_I = 10$       (b) Y-axis  $K_P = 20,000$  &  $K_I = 200$       (c) Z-axis  $K_P = 15,000$  &  $K_I = 10$       (d) Z-axis  $K_P = 20,000$  &  $K_I = 200$       =

Figure A.1: Reaction Wheel Calibration Response -0.25 N-m Commanded Torque

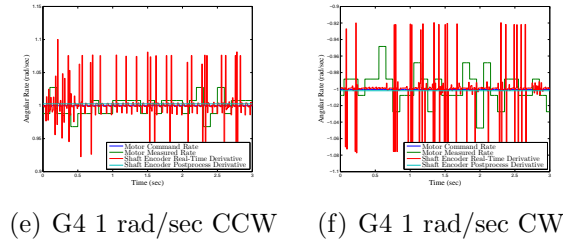
### 1.2 Vehicle MOI

No additional figures.

### 1.3 CMG Gimbal Calibration



(a) G2 1 rad/sec CCW      (b) G2 1 rad/sec CW      (c) G3 1 rad/sec CCW      (d) G3 1 rad/sec CW



(e) G4 1 rad/sec CCW      (f) G4 1 rad/sec CW

Figure A.2: Gimbal Motor Calibration Tests

## 1.4 CMG and Reaction Wheel Rest-to-Rest Test Results

### 1.4.1 Reaction Wheel Rest-to-Rest Test Results.

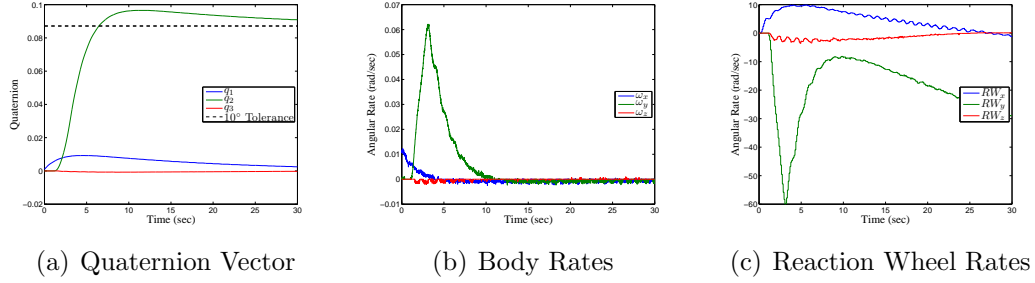


Figure A.3: Reaction Wheel 10° Y-axis Slew

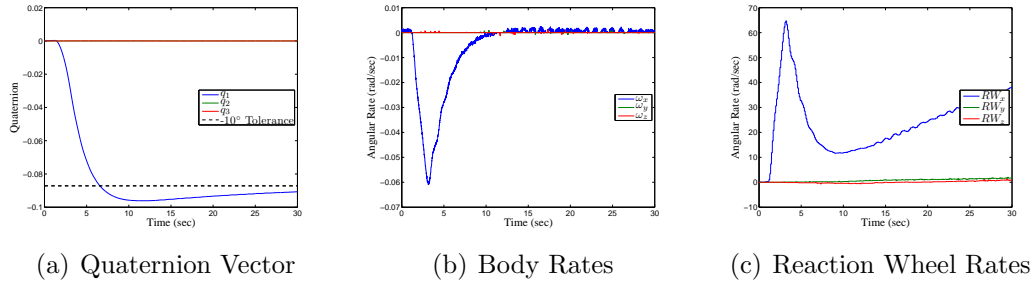


Figure A.4: Reaction Wheel -10° X-axis Slew

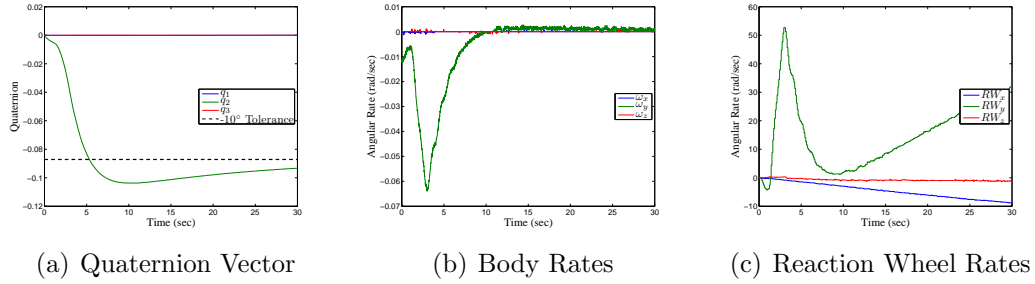


Figure A.5: Reaction Wheel -10° Y-axis Slew

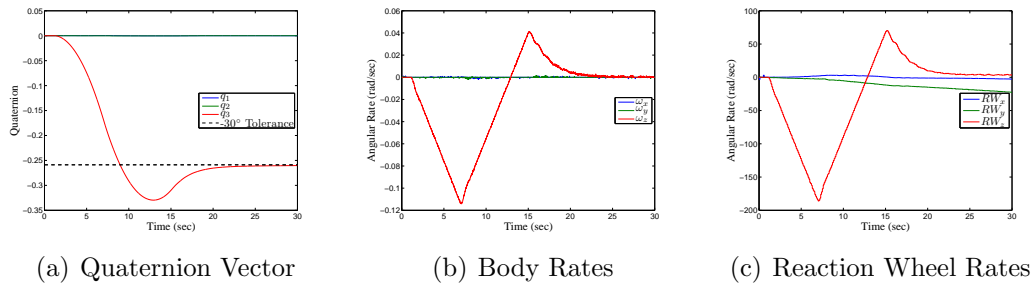


Figure A.6: Reaction Wheel -30° Z-axis Slew

### 1.4.2 CMG Rest-to-Rest Test Results.

#### 1.4.2.1 Y-axis Positive Tests.

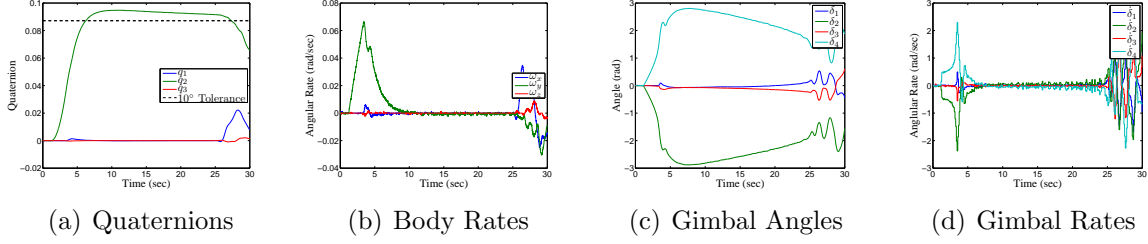


Figure A.7: CMG 10° Y-axis Slew, MPPSL,  $\bar{\delta}_0 = [0, 0, 0, 0]$

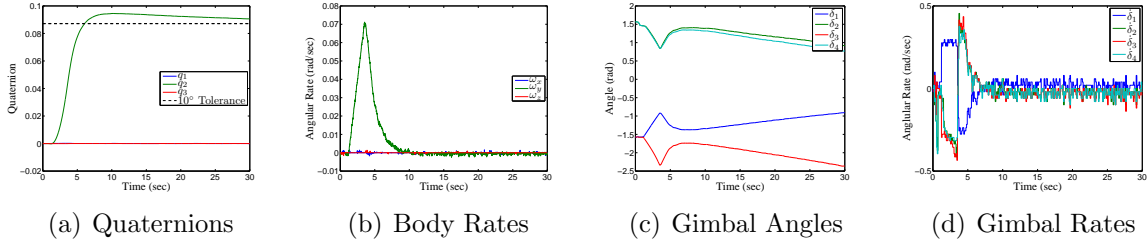


Figure A.8: CMG 10° Y-axis Slew, MPPSL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

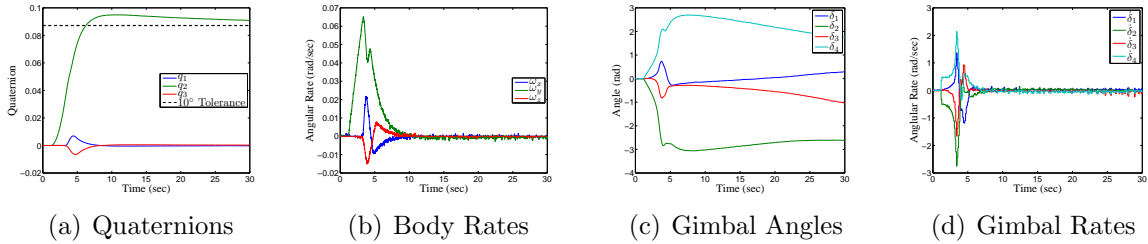


Figure A.9: CMG 10° Y-axis Slew, GISL,  $\bar{\delta}_0 = [0, 0, 0, 0]$

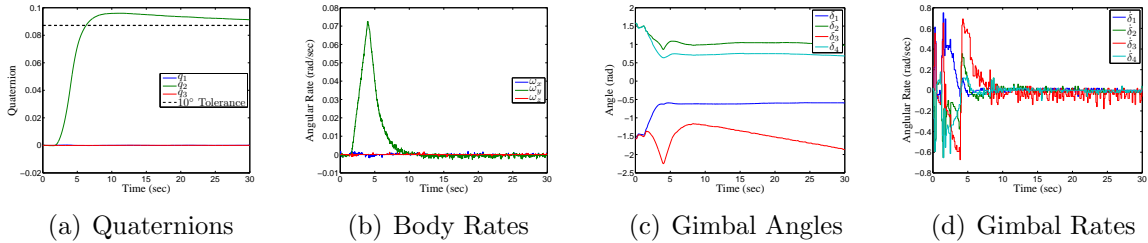


Figure A.10: CMG 10° Y-axis Slew, GISL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

### 1.4.2.2 X-axis Negative Tests.

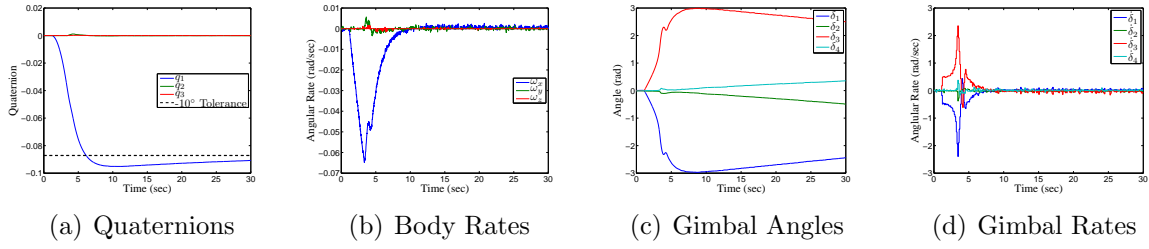


Figure A.11: CMG -10° X-axis Slew, MPPSL,  $\bar{\delta}_0=[0,0,0,0]$

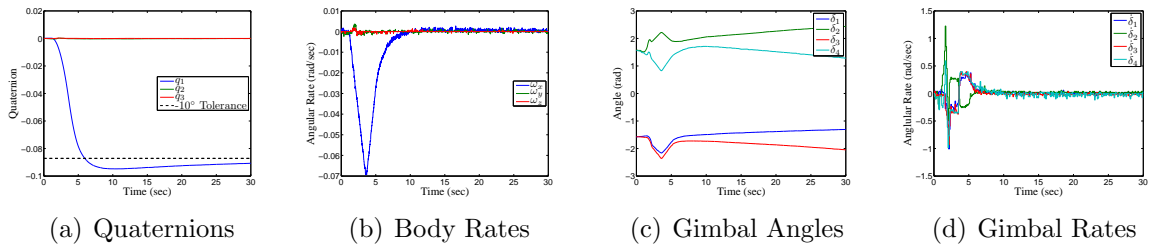


Figure A.12: CMG -10° X-axis Slew, MPPSL,  $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

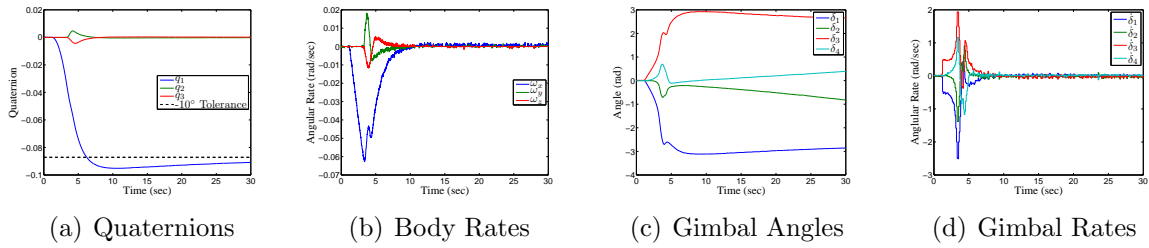


Figure A.13: CMG -10° X-axis Slew, GISL,  $\bar{\delta}_0=[0,0,0,0]$

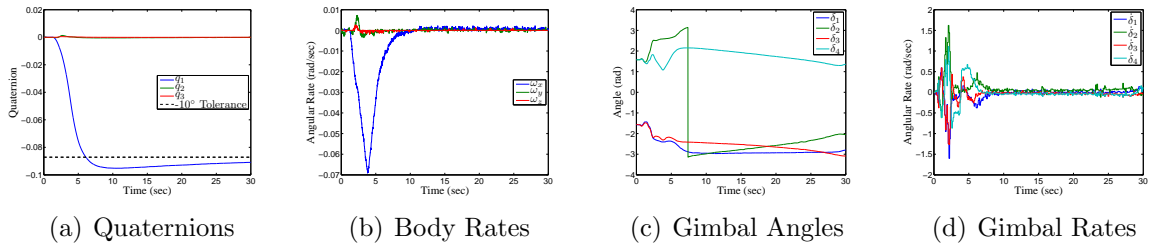


Figure A.14: CMG -10° X-axis Slew, GISL,  $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

### 1.4.2.3 Y-axis Negative Tests.

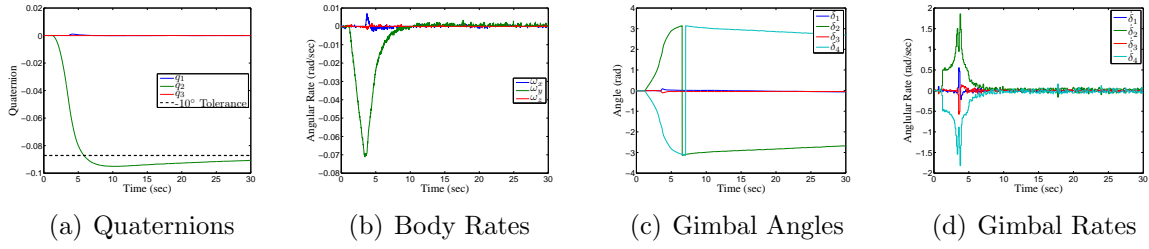


Figure A.15: CMG -10° Y-axis Slew, MPPSL,  $\bar{\delta}_0=[0,0,0,0]$

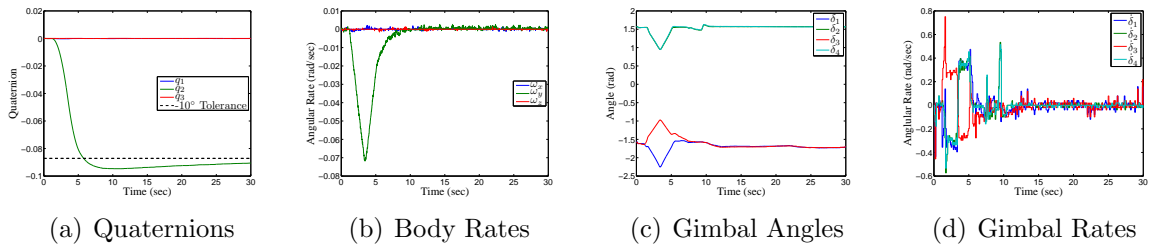


Figure A.16: CMG -10° Y-axis Slew, MPPSL,  $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

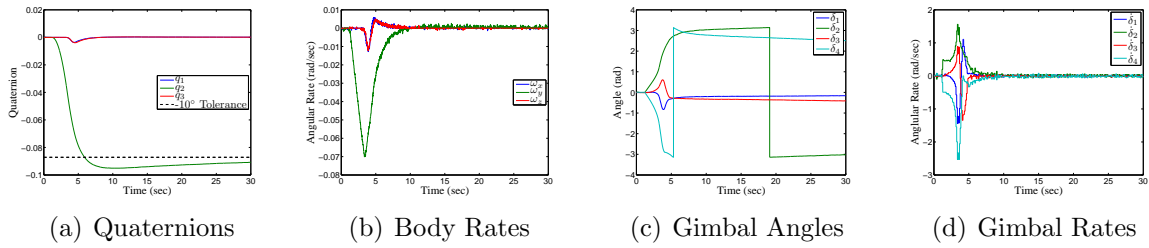


Figure A.17: CMG -10° Y-axis Slew, GISL,  $\bar{\delta}_0=[0,0,0,0]$

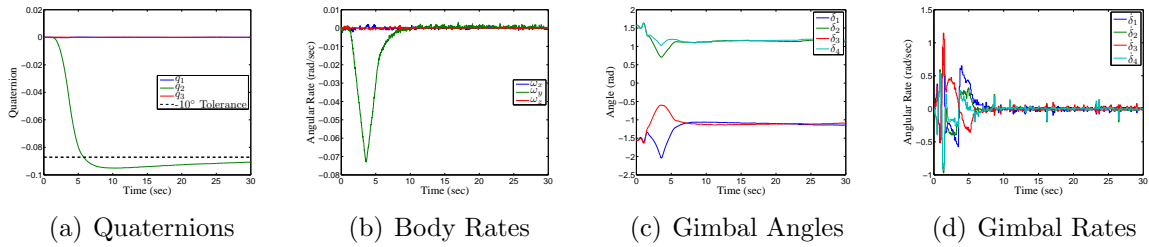


Figure A.18: CMG -10° Y-axis Slew, GISL,  $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

#### 1.4.2.4 Z-axis Negative Tests.

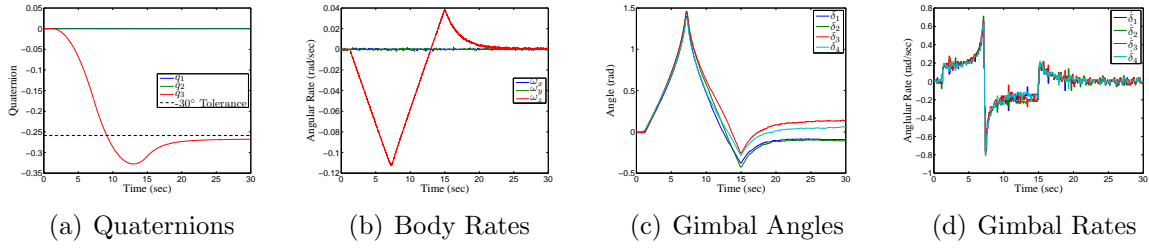


Figure A.19: CMG -30° Z-axis Slew, MPPSL,  $\bar{\delta}_0 = [0, 0, 0, 0]$

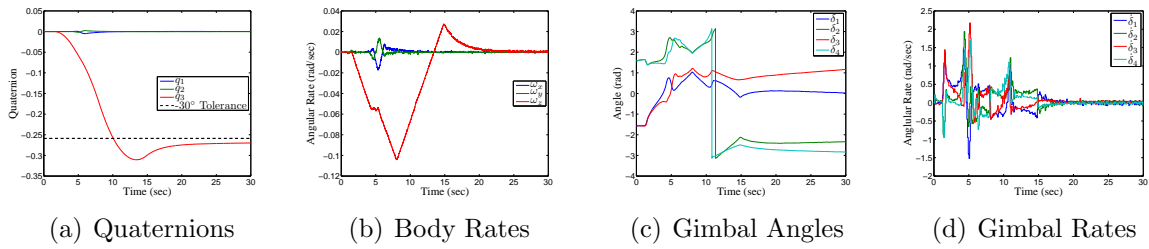


Figure A.20: CMG -30° Z-axis Slew, MPPSL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

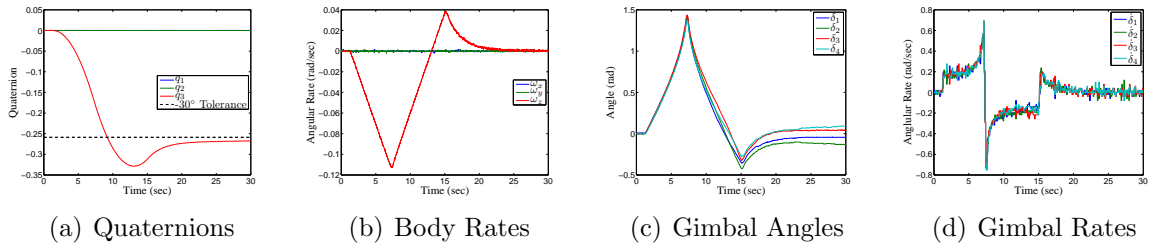


Figure A.21: CMG -30° Z-axis Slew, GISL,  $\bar{\delta}_0 = [0, 0, 0, 0]$

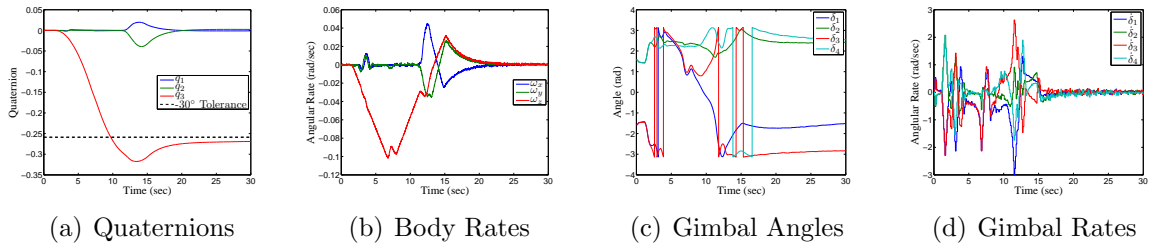


Figure A.22: CMG -30° Z-axis Slew, GISL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$

## 1.5 CMG and Reaction Wheel Torque Test Results

### 1.5.1 Reaction Wheel Torque Test Results.

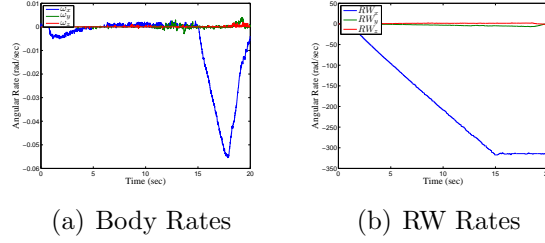


Figure A.23: Reaction Wheel Disturbance Test  $X$ -axis with 33.8 g

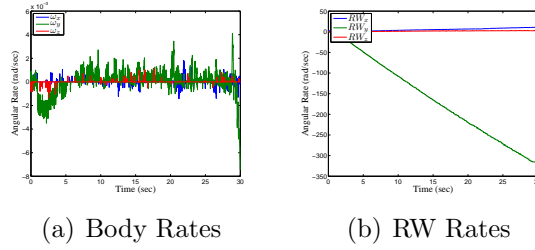


Figure A.24: Reaction Wheel Disturbance Test  $Y$ -axis with 17.2 g

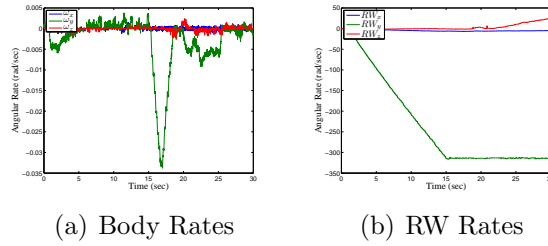


Figure A.25: Reaction Wheel Disturbance Test  $Y$ -axis with 33.8 g

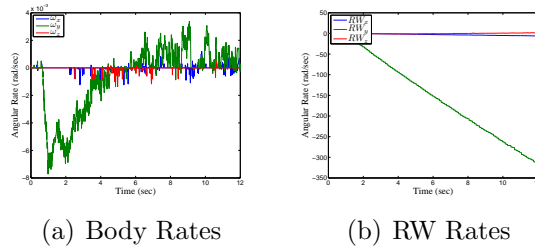
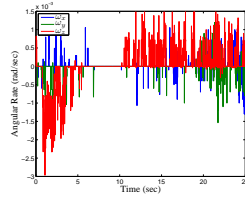
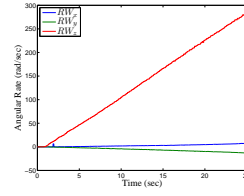


Figure A.26: Reaction Wheel Disturbance Test  $Y$ -axis with 42.6 g

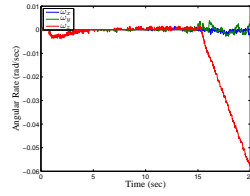


(a) Body Rates

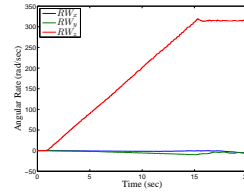


(b) RW Rates

Figure A.27: Reaction Wheel Disturbance Test  $Z$ -axis with 17.2 g

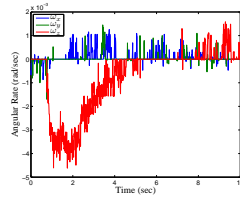


(a) Body Rates

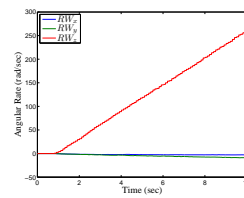


(b) RW Rates

Figure A.28: Reaction Wheel Disturbance Test  $Z$ -axis with 33.8 g



(a) Body Rates



(b) RW Rates

Figure A.29: Reaction Wheel Disturbance Test  $Z$ -axis with 42.6 g



### 1.5.2 CMG Torque Test Results.

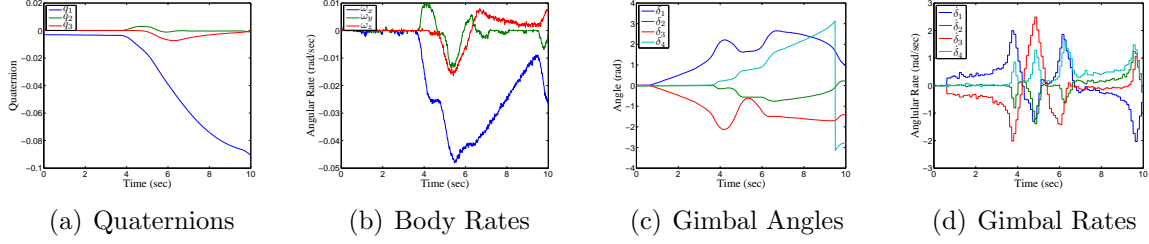


Figure A.30: CMG Disturbance Test, X-axis MPPSL,  $\bar{\delta}_0=[0,0,0,0]$ , 33.8 g

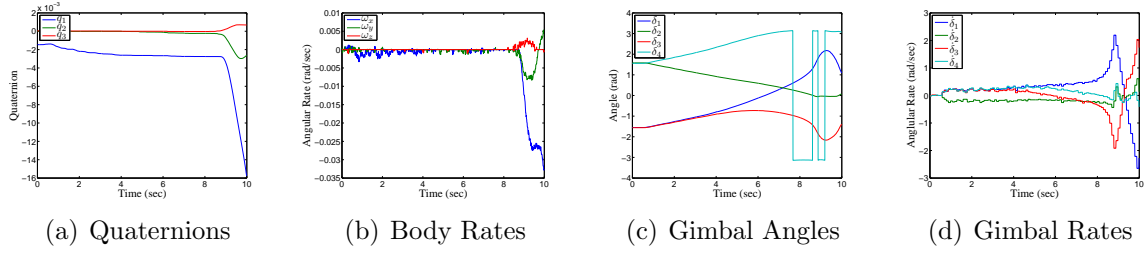


Figure A.31: CMG Disturbance Test, X-axis MPPSL,  $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 33.8 g

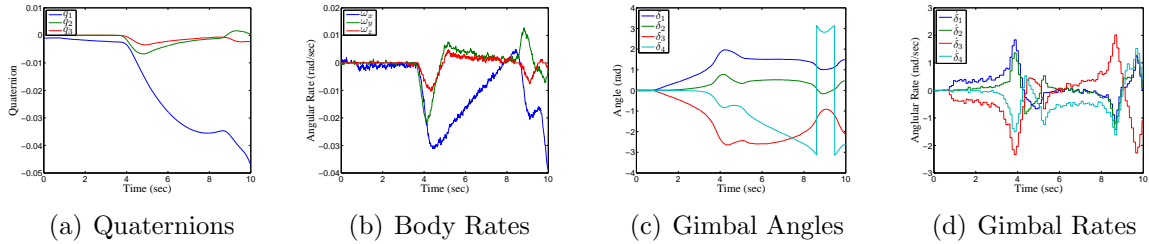


Figure A.32: CMG Disturbance Test, X-axis GISL,  $\bar{\delta}_0=[0,0,0,0]$ , 33.8 g

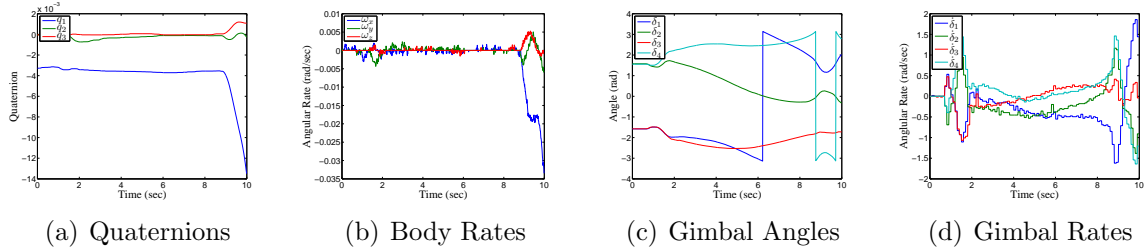


Figure A.33: CMG Disturbance Test, X-axis GISL,  $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 33.8 g

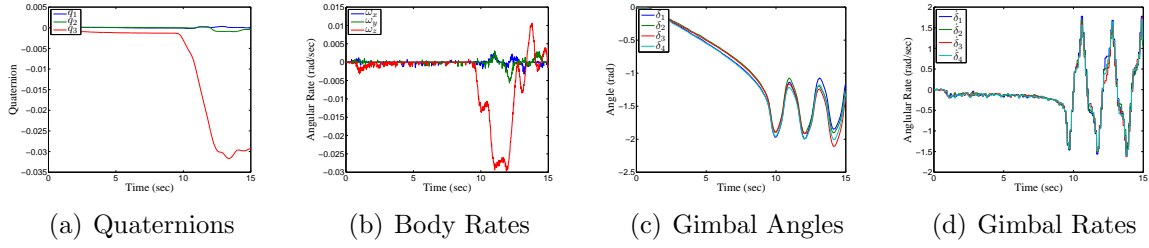


Figure A.34: CMG Disturbance Test, Z-axis MPPSL,  $\bar{\delta}_0=[0,0,0,0]$ , 33.8 g

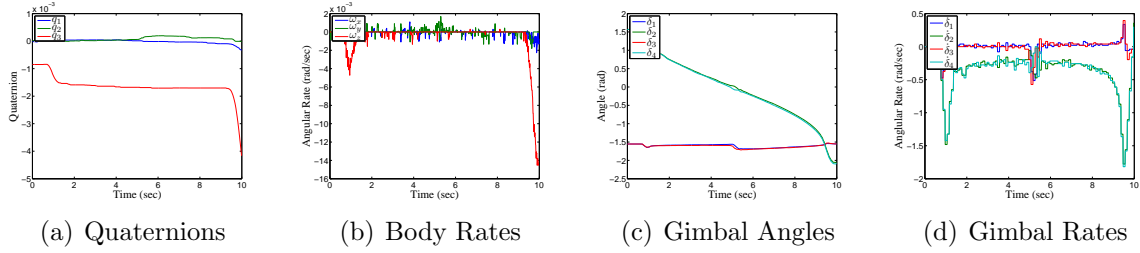


Figure A.35: CMG Disturbance Test, Z-axis MPPSL,  $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 33.8 g

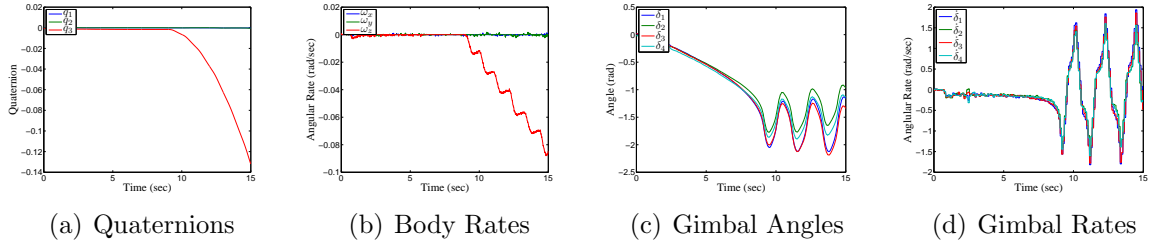


Figure A.36: CMG Disturbance Test, Z-axis GISL,  $\bar{\delta}_0=[0,0,0,0]$ , 33.8 g

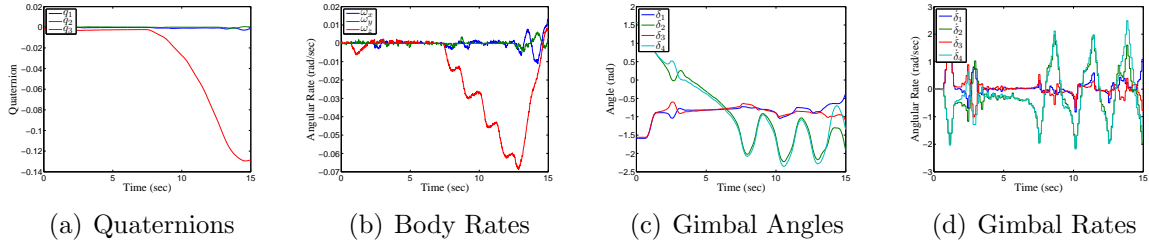


Figure A.37: CMG Disturbance Test, Z-axis GISL,  $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 33.8 g

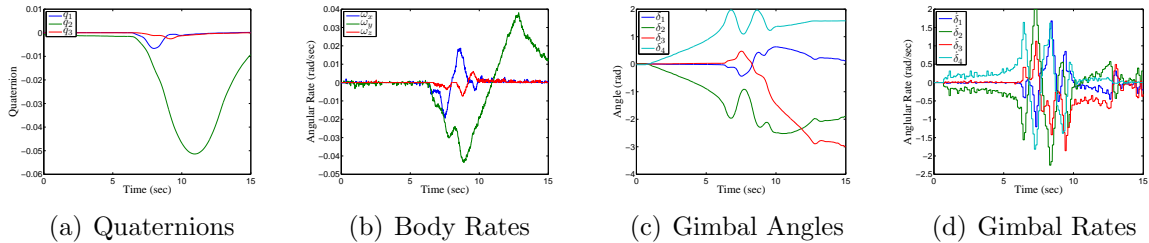


Figure A.38: CMG Disturbance Test, Y-axis MPPSL,  $\bar{\delta}_0=[0,0,0,0]$ , 17.2 g

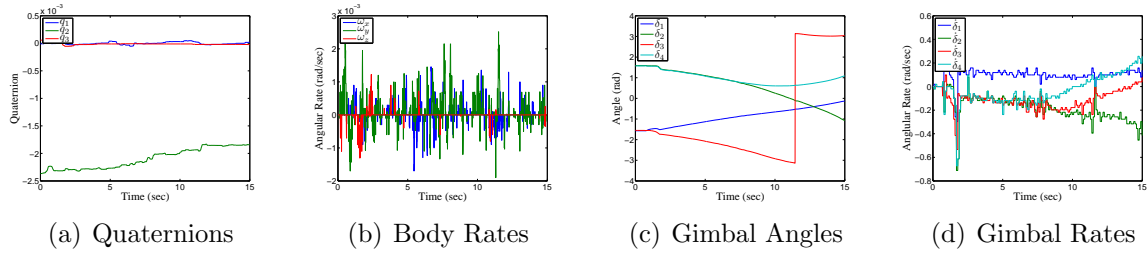


Figure A.39: CMG Disturbance Test, Y-axis MPPSL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 17.2 g

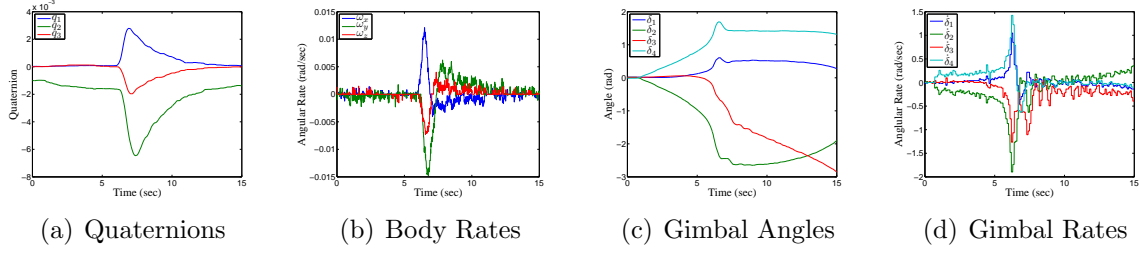


Figure A.40: CMG Disturbance Test, Y-axis GISL,  $\bar{\delta}_0 = [0, 0, 0, 0]$ , 17.2 g

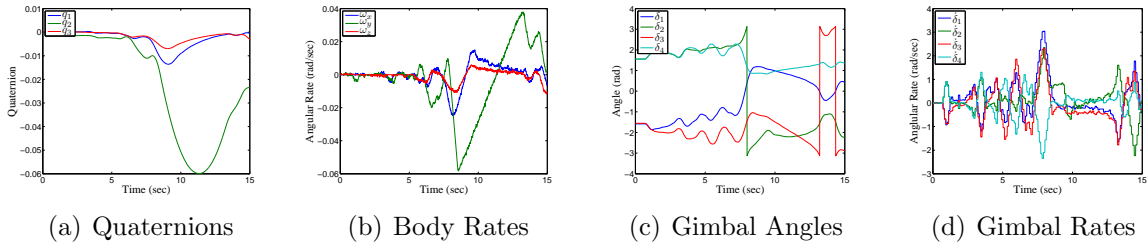


Figure A.41: CMG Disturbance Test, Y-axis GISL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 17.2 g

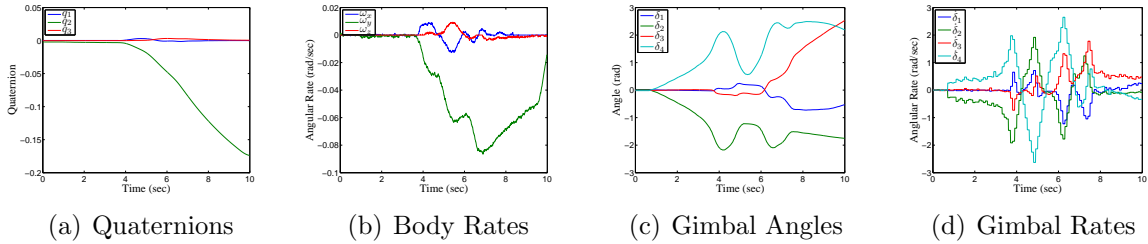


Figure A.42: CMG Disturbance Test, Y-axis MPPSL,  $\bar{\delta}_0 = [0, 0, 0, 0]$ , 33.8 g

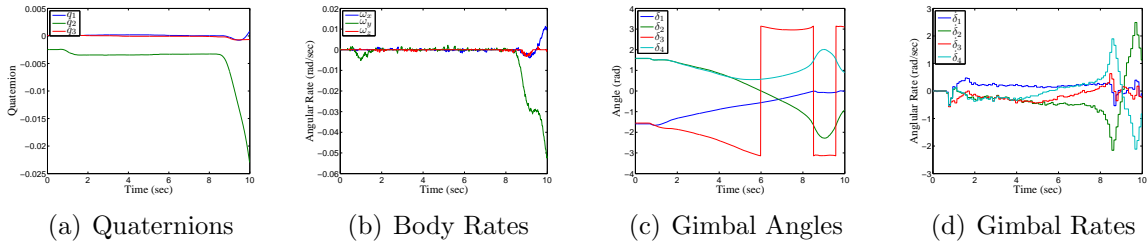


Figure A.43: CMG Disturbance Test, Y-axis MPPSL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 33.8 g

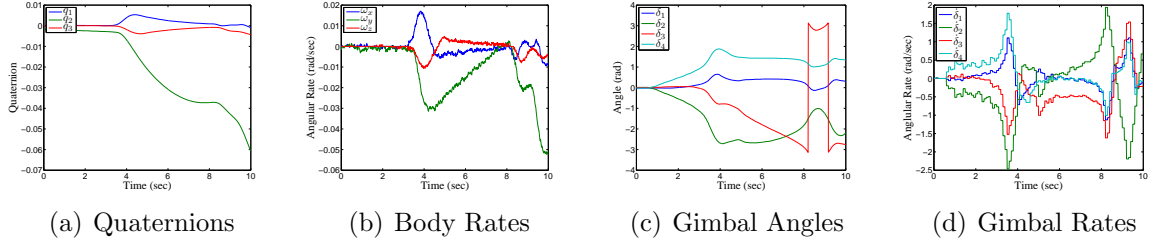


Figure A.44: CMG Disturbance Test, Y-axis GISL,  $\bar{\delta}_0=[0,0,0,0]$ , 33.8 g

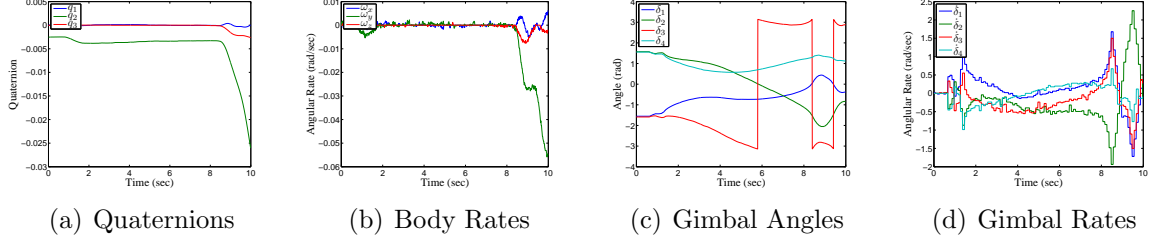


Figure A.45: CMG Disturbance Test, Y-axis GISL,  $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 33.8 g

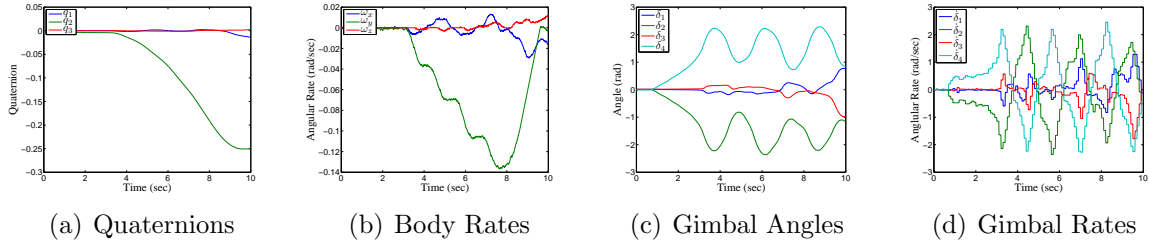


Figure A.46: CMG Disturbance Test, Y-axis MPPSL,  $\bar{\delta}_0=[0,0,0,0]$ , 42.6 g

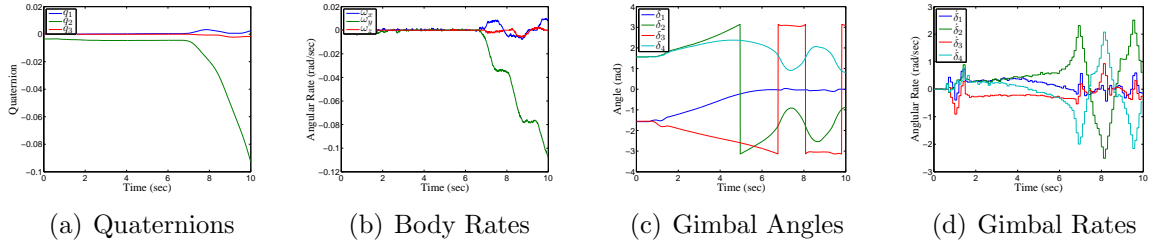


Figure A.47: CMG Disturbance Test, Y-axis MPPSL,  $\bar{\delta}_0=[\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 42.6 g

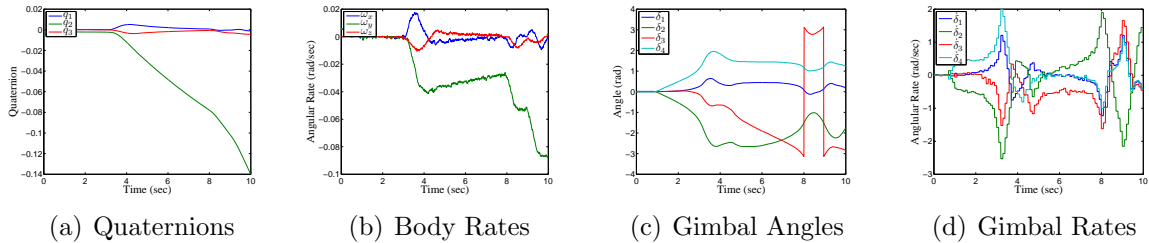


Figure A.48: CMG Disturbance Test, Y-axis GISL,  $\bar{\delta}_0=[0,0,0,0]$ , 42.6 g

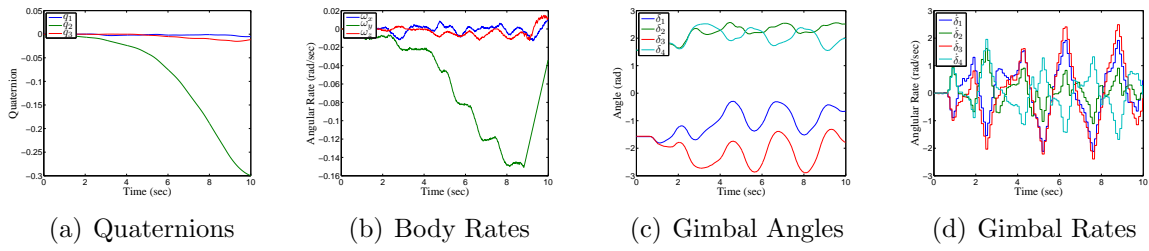
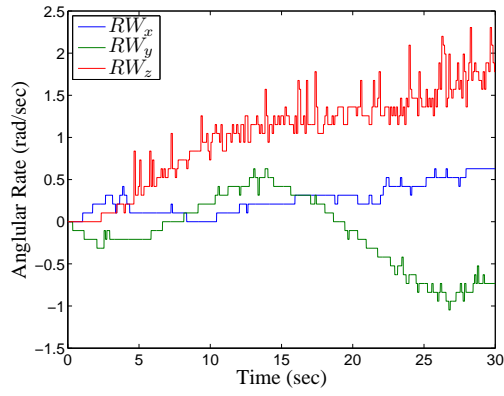
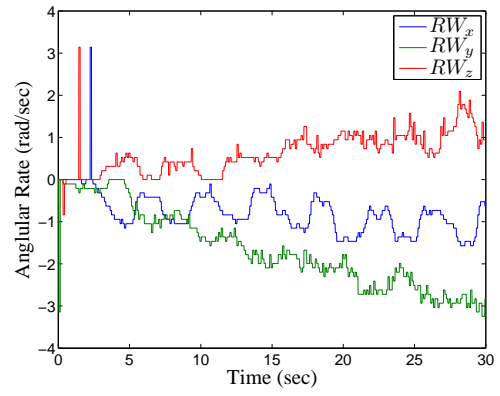


Figure A.49: CMG Disturbance Test, Y-axis GISL,  $\bar{\delta}_0 = [\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}]$ , 42.6 g

## 1.6 CMG Null Motion Test Results



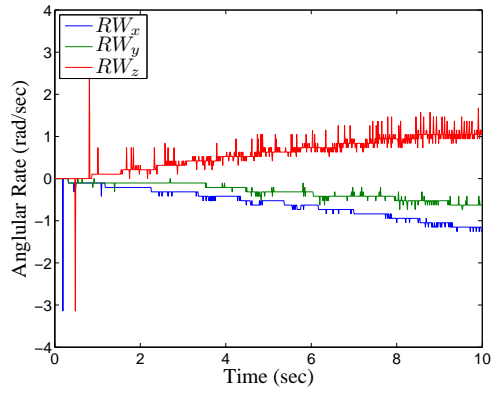
(a) Reaction Wheel Calibration



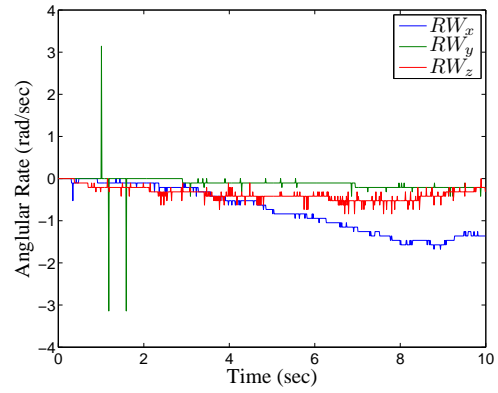
(b) Reaction Wheel Uncorrected

Figure A.50: Null Motion Reaction Wheel Measurements

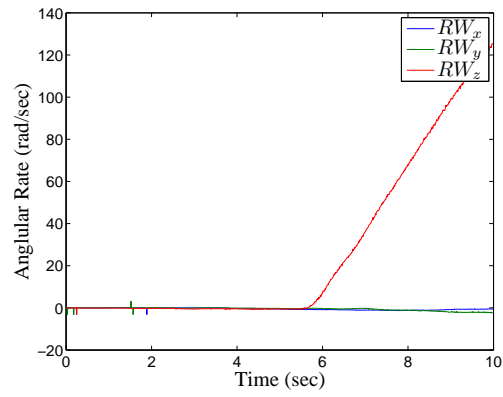
## 1.7 CMG Torque Multiplication Test Results



(a) Reaction Wheel Calibration



(b) Reaction Wheel Gimbals Only



(c) Reaction Wheel CMGs

Figure A.51: Torque Multiplication Reaction Wheel Measurements

## *Appendix B. SimSat Operating Manual*

This appendix provides instructions, information, and tips for operating Simsat including the SIMULINK and MATLAB mLib programs developed for the experiments conducted during this research.

### **2.1 *RTICAN Blockset in Simulink***

This section will outline how to build CANOpen blocks using the RTICAN blockset. The EPOS Firmware and Baumer Firmware documents provide the CANOpen object library address and format for the communications, and should be reviewed prior to altering the CANOpen blocks or adding new blocks.

Data is placed on the CAN bus using the CAN Write block. This block can be found in the RTICAN blockset within the simulink browser. Once added to the program, the configuration can be accessed by double-clicking the block. Figure 1(a) shows the first configuration page, the message tab. The message format type should be ‘STD’, and the node ID specified in hexadecimal based on the communications type and node number. As an example, SDO write messages are addressed to hexadecimal  $0x600 + \text{NodeID}$ , therefore an SDO message to Node 4 would be addressed to  $0x604$ , while a message to node (decimal) 21 would be addressed  $0x615$ .

Tab 2, the message composition tab, is used to configure the message format. CAN messages can carry up to 64 bits of data for a single message. Messages are formatted based on the data to be placed on the bus. For example the SDO message format is:

1. Command Byte - 8 bits
2. Object Library Index - 16 bits
3. Object Library Subindex - 8 bits
4. Data Field - 32 bits

Figure 1(b) shows the standard SDO configuration. Each data item requires specifying signal information. Start bit is calculated based on the previous number of allocated bits and calculated using zero indexing. Signal length is the number of bits for that



item, typically 8, 16, or 32 bits corresponding to 1, 2, or 4 bytes, respectively. Signal type should be ‘Standard.’ For SDO messages, data type should be specified as unsigned integers. For PDO messages, data type should be specified based on the data used, most commonly signed or unsigned integers. Multi-byte items, such as the Data Field or Object Library Index should have byte layout specified as ‘Little endian’ to match the CANOpen protocol. PDO messages are configured in a similar manner, but matching the PDO message format used. As an example, the default PDO for the fan actuators is two signed integer velocities, therefore the message format is two 32 bit signed integers. The EPOS and Baumer Firmware documents list the specific message format information.

The remaining tabs contain addition configuration options, such as message triggering and timing controls. Consult the RTICAN Blockset help file for additional information.

**(a) Message Tab 1**

RTICAN Transmit Message

Message | Message Composition | Options | Custom Encoding

Purpose: Defines a CAN transmit message (TX).

CAN controller specification:

Module type: **CAN Type 1** Controller number: **1**

Module number: **1** Select CAN Controller...

Data file support:

☒ Parameterize block by hand.

☐ Load parameters from data file: <no datafile support> Load Message

TX message parameters:

Message name: **SDO SE 21 TX** Block sample time: **-1** sec

☒ Use as block name

Start time: ☐ set to: **0** sec

Message identifier: ☒ STD ☐ dec ☐ hex **0x615**

Auto repeat time: ☐ set to: **0.1** sec

Message length: **8** Bytes

Timeout: ☐ set to: **0.1** sec

Comments:

OK Cancel Help Apply

**(b) Message Composition Tab 2**

RTICAN Transmit Message

Message | Message Composition | Options | Custom Encoding

Message Name: **SDO SE 21 TX** Id: **0x615** Length: **8** Bytes

Message composition

List signals for mode: **all modes**

Inp	Signal name	SB	SL	ST	MV	DT	BL	Factor	Offset
1	Byte 0 - Command Byte	0	8	S		U	LE	1	0
2	Byte 1-2 - Index	8	16	S		U	LE	1	0
3	Byte 3 - Sub-Index	24	8	S		U	LE	1	0
4	Byte 4-7 - Data	32	32	S		U	LE	1	0

New Copy Delete Comments...

Signal editor

Signal name: **Byte 3 - Sub-Index** Data type: **unsigned int**

Start bit: **24** Byte layout: **little endian**

Signal length: **8** bits Factor: **1**

Signal type: **standard** Offset: **0**

Mode value:  Physical unit:

OK Cancel Help Apply

(a) Message Tab 1

(b) Message Composition Tab 2

Figure B.1: RTICAN Write Block

CANOpen reads are performed using the RTICAN read block, available in the RTICAN library. CANOpen reads are configured in the same way as RTICAN write block by specifying the node address in hexadecimal and configuring the message format.

Lastly, remote transmission request (RTR) messages are sent using the RTICAN request message (RQ) block, not the RTICAN write block. RTRs are addressed to the same node ID as the PDO they trigger. The RTICAN RQ configuration is similar to the read and write blocks. Figure B.2 shows the RTR message for PDO #1 of node 1 addressed as 0x180 for PDO #1 plus 0x001 for node 1.

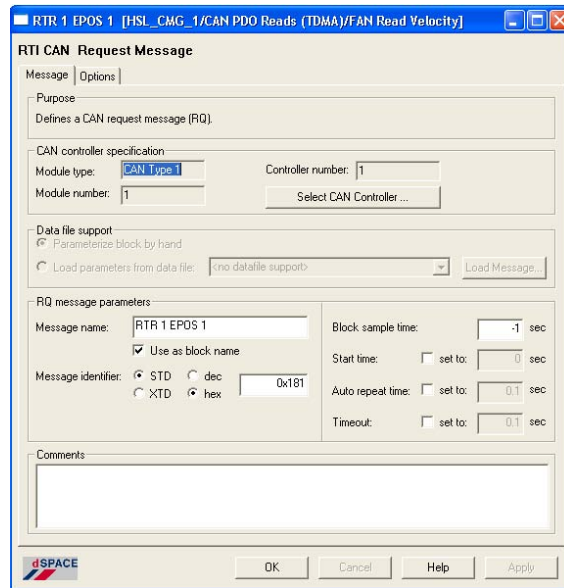


Figure B.2: RTICAN Request Message Block

## 2.2 Simulink *Tips*

Consult the dSpace RTI manuals for explicit instructions on what can, and cannot be include in the simulink real-time programs for the MicroAutoBox. Known items that cannot be included are:

1. MATLAB specific programs or functions such as null()
2. MATLAB \*.mex compiled programs
3. MATLAB user defined functions and scripts.

Most of these capabilities can be provided in SIMULINK using the available blocksets. Importantly, the Signal Processing Blockset and Aerospace Blockset contain several matrix processing algorithms that are useful when developing control laws.

One way to significantly reduce compile time and provide code reuse is to build reusable subroutines or converting duplicate subroutines to a ‘single’ reusable subroutine. Known items that are not allowed inside reusable subroutines are:

1. S-functions (consult dSpace RTI manuals for more information)
2. RTICAN communications blocks
3. Triggered subroutines
4. Enabled subroutines.

One important caveat is that both triggered and enabled subroutines can be converted into reusable subroutines, but they may not contain any triggered or enabled subroutines. To convert a subroutine into a reusable subroutine:

1. Right-click on the subroutine block
2. Select ‘Subsystem Parameters...’
3. The Subsystem configuration screen should open (Fig B.3)
4. Select ‘Treat as atomic unit’ checkbox
5. For the ‘RTW system code:’ pull-down select ‘Reusable Function’
6. For the ‘RTW function name options:’ pull-down select ‘User specified’
7. For ‘RTW function name:’ provide a C-code compatible name. Do not use the exact same name as the subroutine
8. For ‘RTW file name options:’ pull-down select ‘Use function name’

After generating the first subroutine, simply copy and paste to create additional versions. Do not nest reusable subroutines within themselves. Additionally, if changes are required to a reusable subroutine, modify only one. The delete the old versions and replace them with copies of the updated version to prevent SIMULINK from detecting multiple versions with the same file name.

Finally, remember that all variables and blocks within SIMULINK are accessed in ControlDesk and mLib by their label under their block. There are several key exceptions.

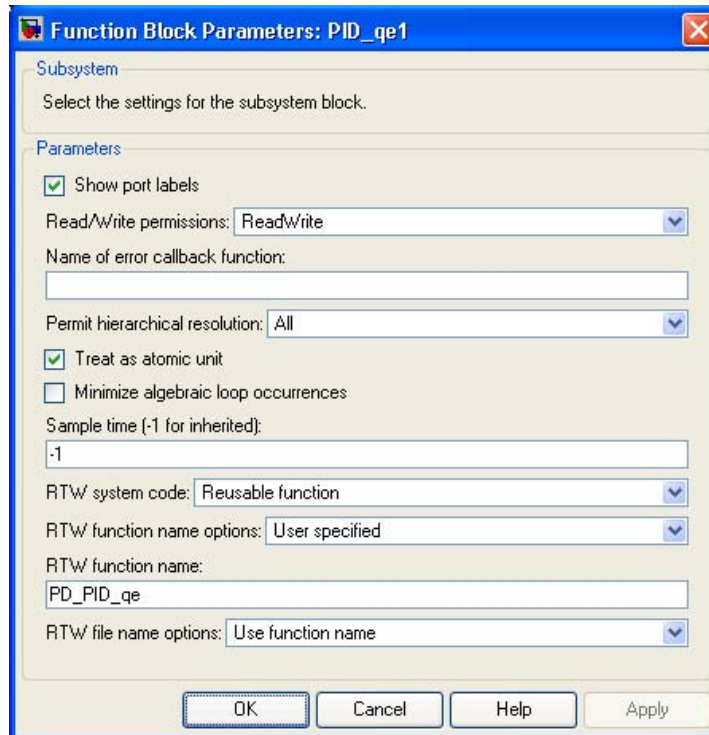


Figure B.3: Subsystem Parameters

First, ‘Data Stores’ can be accessed via their Data Store name (in block name), and via their label. The label controls the initial condition, while the Data Store name is the actual data. The other exception are subsystem in-ports and out-ports do not always appear in ControlDesk and mLib as they will be in-lined by the compiler for efficiency.

### 2.3 *mLib Tips*

The mLib functions are used to access the MicroAutoBox through MATLAB rather than using the ControlDesk interface. The dSpace ControlDesk help file contains instructions on using mLib, this section will outline some tips and undocumented features.

The first step to any mLib program is to select the operating board. On SimSat, the command is “mLib(‘SelectBoard’,‘DS1401’);” and is only required for the first mLib program that is run; however, multiple function calls will not cause a problem.

The next step is to configure the variables for mLib to access. The first step is to generate a column (use “;” not “,”) structure (use “{ and }” not “

*and*

”) of names for the variables. The easiest way to populate this structure is to copy-paste from ControlDesk, which is done by selecting the variable from ControlDesk’s item, shown in Fig B.4, and pressing ‘Ctrl-C’ to copy the variable string. Note that there is no pull-down menu option for this. Once the field has been copied, return to MATLAB and paste the variable string. I suggest building at least two variable structures with one specifically for data capture.

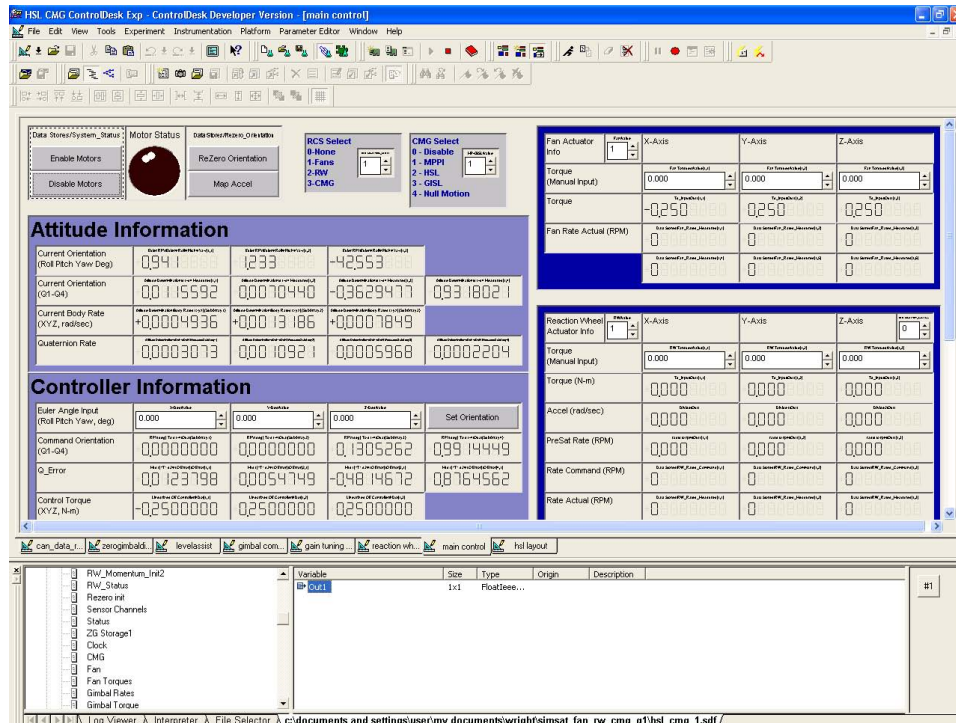


Figure B.4: Control Desk Variable Selection

After building the variable list structures, the variables must be referenced for access via mLib using the ‘GetTrcVar’ mLib command. Do not use the same variable name as the list. The command “controlVar = mlib(‘GetTrcVar’,controlName);” creates the variable structure ‘controlVar’ to match the variable names specified in ‘controlName’ which is the naming convention I used.

Once the variables are setup, reading data from the MicroAutoBox is done using the command `“data = mlib(‘Read’,controlVarx)”` where ‘x’ corresponds to the variable in the list to be read. The information will be returned to ‘data’ with the format specified by the Simulink Model (scalar, row vector, column vector, matrix). If more than one item is selected, then mLib will return a structure with the contents matching the Simulink Model.

Writing to variables is slightly more complicated. Writing to a single variable is performed using the command `“mlib(‘Write’,controlVarx,‘Data’,[1,2,3])”` where ‘x’ specifies the single entry to write to, and ‘[1,2,3]’ is the data to be written. In this example, the specified variable is a 1x3 row vector. If you do not match the dimensions, mLib will throw an error during execution. In order to write multiple variables, mLib requires a data structure. For example `“mlib(‘Write’,controlVar1,2,‘Data’,1,[2;3;4])”` where ‘controlVar1’ accepts a scalar, and ‘controlVar2’ accepts a 3x1 column vector. If the structure and variable dimensions do not match then mLib will throw an error message.

Capturing data using mLib requires the use of the ‘Set’ command. There are several options for configuring data captures which are covered in the ControlDesk help file, so I will focus on tips. First, it is significantly easier to specify capture variables as their own variable list, rather than trying to select from a master list of variables. Next, remember that time ‘runs right’ and variables ‘run down’ per the capture list, with the first row as the time stamp if ‘TimeStamping’ is enabled. The length of the capture will be the number of samples times ‘DownSampling’ times step size of 1 ms. Capture progress/status can be monitored using the `“mLib(‘CaptureStatus’)”` and `“mLib(‘TriggerStatus’)”` for triggered captures.

Finally, while the `“pause(x)”` command will pause MATLAB execution for a given time, a better option is to reference the MicroAutoBox clock using mLib read the clock signal. This will ensure that MATLAB remains synchronized.

## Bibliography

1. Richard Boynton. “Using A Spherical Air Bearing to Simulate Weightlessness,”. In *55th Annual Conference of the Society of Allied Weight Engineers*. SAWE, June 1996.
2. James E. Colebank, Robert D. Jones, George R. Nagy, Randall D. Pollack, and Donald R. Mannebach. *SIMSAT: A Satellite System Simulator and Experimental Test Bed for Air Force Research*. MS thesis, Air Force Institute of Technology (AU), March 1999. ADA361635.
3. Vincent J. Dabrowski. *Experimental Demonstration of an Algorithm to Detect the Presence of a Parasitic Satellite*. MS thesis, Air Force Institute of Technology (AU), March 2003. (ADA413233).
4. Ball Aerospace DigitalGlobe. Digitalglobe fact sheet: Worldview-2.
5. dSPACE. *ControlDesk “Experiment Guide”*, Release 4.2 — March edition, 2005.
6. dSPACE. *ControlDesk “Instrument Reference”*, Release 4.2 — March edition, 2005.
7. Leonhard Euler. General formulas for the translation of an arbitrary rigid body. Translation of original text published by Johan Sten.
8. David B. French. *Hybrid Control Strategies for Rapid, Large Angle Satellite Slew Maneuvers*. MS thesis, Air Force Institute of Technology (AU), March 2003. ADA412869.
9. Joseph M. Fulton. *Attitude Control and Multimedia Representation of Air Force Institute of Technology’s (AFIT’S) Simulation Satellite (SIMSAT)*. MS thesis, Air Force Institute of Technology (AU), March 2000. ADA380275.
10. Jason W. Geitgey. *The Determination of Remaining Satellite Propellant Using Measured Moments of Inertia*. MS thesis, Air Force Institute of Technology (AU), June 2006. ADA453964.
11. Michael S. Hines. *Fuel Estimation Using Dynamic Response*. MS thesis, Air Force Institute of Technology (AU), March 2007. ADA469309.

12. Peter C Hughes. *Spacecraft Attitude Dynamics*. Dover Publications, Inc, 31 East 2nd Street, Mineola, NY 11501, 2nd edition, 2004, 1986.
13. Weiping Li Jean-Jacques Slotine. *Applied Nonlinear Control*. Prentice-Hall International, Upper Saddle River, New Jersey 07458, 1991.
14. ByungMoon Kim, Efstathios Velenis, Patrick. Kriengsiri, and Panagiotis Tsiotras. "A Spacecraft Simulator for Research and Education,". In *Astrodynamics Specialists Conference*, pages 897–914. AIAA/AAS, July 2001.
15. Matthew B. Kimsal. *Design of a Space-Borne Autonomous Infrared Tracking System*. MS thesis, Air Force Institute of Technology (AU), March 2004. ADA425222.
16. Frederick A. Leve. *NOVEL STEERING AND CONTROL ALGORITHMS FOR SINGLE-GIMBAL CONTROL MOMENT GYROSCOPES*. PhD thesis, The University of Florida, 2010.
17. Maxon Motor. *EPOS 70/10 "Getting Started"*, december edition, 2008.
18. Maxon Motor. *Maxon EC 45 Order Number 136211 "Catalog Page"*, May edition, 2009.
19. Maxon Motor. *Maxon EC-MAX 30 Order Number 272764 "Catalog Page"*, 2010 edition, 2010.
20. Maxon Motors. *Maxon GP-32 Order Number 326671 "Catalog Page"*, 2010 edition, 2010.
21. C. Douglas McFarland. *Near Real-Time Closed-Loop Optimal Control Feedback for Spacecraft Attitude Maneuvers*. MS thesis, Air Force Institute of Technology (AU), March 2009. ADA496781.
22. NASA. Nasa glenn research center zero gravity facility overview.
23. NASA. Nasa/jsc aircraft operations reduced gravity research program.
24. Katsuhiko Ogata. *Modern Control Engineering*. Prentice-Hall, New Jersey, 4th edition, 2002.



25. Neal R. Roach, Wayne C. Rohe, and Nathan F. Welty. *A Systems Engineering Approach to the Design of a Spacecraft Dynamics and Control Testbed*. MS thesis, Air Force Institute of Technology (AU), March 2008.
26. I. Michael Ross and Fariba Fahroo. “Issues in the Real-Time Computation of Optimal Control”. *Mathematical and Computer Modelling: An International Journal*, 43, 2005.
27. I. Michael Ross, Pooya Sekhavat, Andrew Fleming, and Qi Gong. “Optimal Feedback Control: Foundations, Examples, and Experimental Results for a New Approach”. *Journal of Guidance, Control, and Dynamics*, 31(2):307–321, 2008.
28. P. L. Palmer S. Asghar and M. Roberts. Exact steering law for pyramid-type four control moment gyro systems. In *AIAA/AAS Astrodynamice Specialist Conference*, 2006.
29. George Sagnac. “Sur la Preuve de la Réalité de l’Ether Lumineux par l’Expérience de l’Interférographe Tournant”. *Comptes Rendus*, 157:1410–1413, December 1913.
30. John Schaub, Hanspeter & Junkins. *Analytical Mechanics of Space Systems*. AIAA Education Series. AIAA Education Series, 2nd edition, 2009.
31. Jana L. Schwartz, Mason A. Peck, and Christopher D. Hall. “Historical Review of Spacecraft Simulators”. *Journal of Guidance, Control, and Dynamics*, 26(4):513–522, 2003.
32. Jerry J. Sellers. *Understanding Space: An Introduction to Astronautics*. The McGraw-Hill Companies, Inc., New York, 3rd edition, 2005.
33. Jason E. Smith. *Attitude Model of a Reaction Wheel/Fixed Thruster Based Satellite Using Telemetry Data*. MS thesis, Air Force Institute of Technology (AU), March 2005. ADA437523.
34. Ryan E. Snider. *Attitude Control of a Satellite Simulator Using Reaction Wheels and a PID Controller*. MS Thesis, Air Force Institute of Technology (AU), March 2010.
35. Gilbert Strang. *Linear Algebra and its Applications*. Thompson Corporation, 10 Davis Drive Belmont, CA 94002-3098, 4th edition, 2006.

36. William E. Wiesel. *Modern Astrodynamics*. Aphelion Press, 2652 Yalonda Ct. Beaver creek, OH 45434, 1.05 edition, 2003.
37. Wikipeda. Wikipedia: Iss.

<b>REPORT DOCUMENTATION PAGE</b>			Form Approved OMB No. 0704-0188	
The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.				
1. REPORT DATE (DD-MM-YYYY) 24-03-2011		2. REPORT TYPE Master's Thesis		3. DATES COVERED (From — To) Aug 2009 – Mar 2011
4. TITLE AND SUBTITLE Design Of Attitude Control Actuators for a Simulated Spacecraft			5a. CONTRACT NUMBER	
			5b. GRANT NUMBER	
			5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)  McChesney, Christopher G., Capt, USAF			5d. PROJECT NUMBER	
			5e. TASK NUMBER	
			5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/ENY) 2950 Hobson Way WPAFB OH 45433-7765			8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GA/ENY/11-M12	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Research Laboratory Space Vehicles Directorate Attn: Capt. Chester McFarland 3550 Aberdeen Ave SE Kirtland AFB, NM 87117 DSN: 263-3638 Email: chester.mcfarland@kirtland.af.mil			10. SPONSOR/MONITOR'S ACRONYM(S) AFRL/RV	
			11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION / AVAILABILITY STATEMENT APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED				
13. SUPPLEMENTARY NOTES This material is declared the work of the U.S. Government and is not subject to copyright protection in the United States.				
14. ABSTRACT The Air Force Institute of Technology's attitude dynamics simulator, SimSat, is used for hardware-in-the-loop validation of new satellite control algorithms. To provide the capability to test algorithms for control moment gyroscopes, SimSat needed a control moment gyroscope array. The goal of this research was to design, construct, test, and validate a control moment gyroscope array for SimSat. The array was required to interface with SimSat's existing structure, power supply, and electronics. The array was also required to meet maneuver specifications and disturbance rejection specifications. First, the array was designed with initial sizing estimates based on requirements and vehicle size. Next, the vehicle and control dynamics were modeled to determine control moment gyroscope requirements and provide a baseline for validation. Control moment gyroscopes were then built, calibrated, and installed on the vehicle. The actuators were then validated against the dynamics model. Testing shows minor deviation from the expected behavior as a result of small misalignments from the theoretical design. Once validation was complete, the array was tested against the performance specifications. The performance tests indicated that the control moment gyroscope array is capable of meeting specifications.				
15. SUBJECT TERMS Attitude Control, Control Moment Gyroscope (CMG), Spacecraft Dynamics, Satellite Simulator				
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT  UU	18. NUMBER OF PAGES  191
a. REPORT  U	b. ABSTRACT  U	c. THIS PAGE  U		
			19a. NAME OF RESPONSIBLE PERSON Dr. Eric D. Swenson	
			19b. TELEPHONE NUMBER (Include Area Code) (937)255-3636, ext 7479 Email: eric.swenson@afit.edu	